



**TUGAS AKHIR - TE 145561**

## **LOKER PENITIPAN OTOMATIS DILENGKAPI SISTEM PEMBAYARAN**

Akbar Riski Darrani  
NRP 2214030011

Dosen Pembimbing  
Ir. Hanny Boedinugroho, MT.

PROGRAM STUDI KOMPUTER KONTROL  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017





**FINAL PROJECT - TE 145561**

***AUTOMATIC DEPOSIT LOCKER WITH PAYMENT  
SYSTEM***

Akbar Riski Darrani  
NRP 2214030011

*Advisor*  
Ir. Hanny Boedinugroho, MT.

***COMPUTER CONTROL STUDY PROGRAM  
Electrical and Automation Engineering Departement  
Faculty of Vocational  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017***



## TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **“LOKER PENITIPAN OTOMATIS DILENGKAPI SISTEM PEMBAYARAN”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 17 Juli 2017



Akbar Riski Darrani  
NRP 2214030011

-----Halaman ini sengaja dikosongkan-----

# **LOKER PENITIPAN OTOMATIS DILENGKAPI SISTEM PEMBAYARAN**

## **TUGAS AKHIR**

Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada  
Program Studi Komputer Kontrol  
Departemen Teknik Elektro Otomasi  
Institut Teknologi Sepuluh Nopember



Menyetujui:  
Dosen Pembimbing

Ir. Hanny Boedinugroho, MT.  
NIP.19610706 198701 1 001

**SURABAYA  
JULI, 2017**

-----Halaman ini sengaja dikosongkan-----



# LOKER PENITIPAN OTOMATIS DILENGKAPI SISTEM PEMBAYARAN

**Nama** : Akbar Riski Darrani  
**Pembimbing** : Ir. Hanny Boedinugroho, MT.

## ABSTRAK

Melihat pada tempat penitipan berupa loker yang ada di tempat-tempat umum saat ini yang masih menggunakan sistem konvensional dimana biasanya masih menggunakan kunci manual sudah dirasa tidak efisien. Karena ketika seseorang menggunakan kunci manual pada loker, maka pengunjung tersebut harus membawa kunci loker tersebut kemanapun dia pergi.

Namun hal tersebut dapat menimbulkan masalah ketika kunci loker yang kita bawa hilang. Dengan adanya kemajuan teknologi yang semakin lama semakin canggih, diciptakan alat penitipan berupa loker berbayar yang bekerja secara otomatis sehingga pengguna tidak perlu lagi membawa kunci loker tersebut. Hal tersebut dikarenakan loker tersebut menggunakan *password* sebagai pengunci, dimana pada saat pengunjung ingin menitipkan barang, pengunjung tersebut diharuskan memasukkan *password* sesuai yang diinginkan, lalu setelah *password* dimasukkan maka loker otomatis terkunci dan waktu serta biaya mulai berjalan. Untuk membuka loker tersebut pengunjung diharuskan terlebih dahulu membayar sesuai lama pengunjungan loker, setelah itu pengunjung memasukkan *password* yang telah dimasukkan sebelumnya. Jika *password* yang dimasukkan sesuai, maka kunci loker otomatis terbuka.

Dari hasil pengujian kinerja alat secara keseluruhan, didapatkan persentase keberhasilan kinerja alat sebesar 87.5 % dan presentase *error* sebesar 12.5%.

**Kata Kunci** : Keamanan loker penitipan barang, sistem kunci otomatis,  
*Password*

-----Halaman ini sengaja dikosongkan-----

## ***AUTOMATIC DEPOSIT LOCKER WITH PAYMENT SYSTEM***

**Name : Akbar Riski Darrani**

**Advisor : Ir. Hanny Boedinugroho, MT.**

### ***ABSTRACT***

*Look at the daycare form lockers that are in public places is currently still using conventional systems where usually still use manual keys already proved inefficient. Because when someone uses a manual lock on my locker, so Visitors must bring the locker key wherever he goes.*

*It can cause problems when key lockers we carry are gone. With the technological advances that the longer the more sophisticated tool, was created in the form of storage lockers paid work automatically so that users no longer need to carry a key to the locker. That is because the locker using the password as the locker, where at the time the visitor wants to deposit items, such visitors are required to enter the password as desired, and then after the password is entered then the lockers automatically locked and time of use the locker began to walk. To open the visitor's lockers are required to first pay the corresponding long recursively lockers, after which the visitors enter the password has been entered before. If the password entered is appropriate, then automatic lockers lock open.*

*From the results of performance testing tools overall, obtained the percentage of success is the performance tool of 87.5% and the percentage error of 12.5%*

**Keywords:** *Security lockers luggage system, automatic lock system, Password*

-----Halaman ini sengaja dikosongkan-----

## **KATA PENGANTAR**

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik. Shalawat serta salam semoga selalu tercurahlimpahkan kepada Nabi Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Bidang Studi Komputer Kontrol, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

### **LOKER PENITIPAN OTOMATIS DILENGKAPI SISTEM PEMBAYARAN**

Penyusunan Tugas Akhir ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu, pada kesempatan ini penulis hendak mengucapkan terimakasih kepada semua pihak yang telah mendukung dan membantu dalam pembuatan Tugas Akhir ini. Ucapan terimakasih penulis sampaikan kepada:

1. Allah SWT atas rahmat serta hidayahNya dalam menyelesaikan Tugas Akhir ini.
2. Kedua orang tua yang telah memberikan dukungan penuh dan doa yang tidak pernah putus.
3. Dosen pembimbing kami, yaitu Bapak Ir. Hanny Boedinugroho, MT. Karena telah bersedia meluangkan waktunya untuk membimbing penulis dari awal hingga akhir pembuatan Tugas Akhir ini.
4. Teman-teman angkatan 2014 atas dukungan, semangat, dan kerja samanya.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 17 Juli 2017

Penulis

-----Halaman ini sengaja dikosongkan-----

# DAFTAR ISI

	HALAMAN
HALAMAN JUDUL.....	i
HALAMAN JUDUL.....	iii
PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
HALAMAN PENGESAHAN .....	vi
ABSTRAK .....	ix
ABSTRACT .....	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR .....	xvii
DAFTAR TABEL .....	xix
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Permasalahan .....	1
1.3 Tujuan .....	2
1.4 Batasan Masalah.....	2
1.5 Metodologi Penelitian .....	2
1.6 Sistematika Laporan.....	3
1.7 Relevansi .....	4
BAB II TEORI DASAR .....	5
2.1 Mikrokontroler ATmega 328 .....	5
2.2 DC Power Supply.....	7
2.3 Solenoid Doorlock.....	9
2.4 Coin Acceptor CH-926.....	9
2.5 LCD (Liquid Crystal Display) 20x4.....	12
2.6 Limit Switch.....	13
2.7 Driver Relay .....	14
2.8 Komunikasi Serial RS232 .....	14
2.9 MicroSD .....	16
2.10 Modul RTC (Real Time Clock) DS 1307 .....	17
2.11 Pemrograman Arduino .....	19
BAB III PERANCANGAN DAN PEMBUATAN ALAT .....	21

3.1	Perancangan <i>Hardware</i> .....	23
3.1.1	Perancangan Mikrokontroler ATmega 328.....	23
3.1.2	Perancangan Rangkaian DC <i>Power Supply</i> .....	26
3.1.3	Perancangan <i>Driver Relay</i> .....	28
3.1.4	<i>Setting Coin Acceptor</i> CH-926 .....	29
3.1.5	Perancangan <i>Connector</i> IC-MAX323 dengan DB9 .....	31
3.2	Perancangan <i>Software</i> .....	33
3.2.1	Program Pembayaran .....	33
3.2.2	Program <i>Input Coin Acceptor</i> ke Mikrokontroler ATmega 328 .....	38
3.2.3	Program Akses Waktu Modul RTC ( <i>Real Time Clock</i> ) ...	39
3.2.4	Program <i>Timer Internal</i> .....	41
3.2.5	Program Pengontrol <i>Driver Relay</i> untuk <i>Solenoid Door Lock</i> .....	43
3.2.6	Program Komunikasi Serial melalui RS232 .....	45
3.3	Desain Loker.....	46
BAB IV HASIL PENGUKURAN DAN PENGUJIAN ALAT .....		51
4.1	Pengukuran Tegangan <i>Output DC Power Supply</i> .....	51
4.2	Pengukuran Tegangan <i>Output</i> pin Mikrokontroler ATmega 328	52
4.3	Pengukuran Tegangan <i>Input</i> yang Diperlukan Pada <i>Solenoid Doorlock</i> .....	52
4.4	Pengujian <i>Timer Internal</i> Mikrokontroler ATmega 328.....	53
4.5	Pengujian Modul RTC ( <i>Real Time Clock</i> ) DS 1307 .....	54
4.6	Pengujian Sensor <i>Coin Acceptor</i> CH-926.....	55
4.7	Pengujian Tampilan Pada <i>Serial Monitor</i> .....	58
4.8	Pengujian Kinerja Alat Keseluruhan.....	58
4.8.1	Fitur <i>Inset Coin</i> .....	59
4.8.2	Fitur <i>Timer</i> dan Tarif Penggunaan .....	60
4.8.3	Fitur <i>Password</i> .....	61
BAB V PENUTUP .....		65
5.1	Kesimpulan .....	65
5.2	Saran .....	65
DAFTAR PUSTAKA .....		67
LAMPIRAN .....		69
A.	Listing Program Keseluruhan .....	69
B.	Flow Chart Program Keseluruhan.....	85
C.	Datasheet.....	89



## DAFTAR GAMBAR

	HALAMAN
<b>Gambar 2.1</b> Mikrokontroler ATmega 328 .....	7
<b>Gambar 2.2</b> <i>Solenoid Doorlock</i> .....	9
<b>Gambar 2.3</b> <i>Coin Acceptor</i> CH-926 .....	10
<b>Gambar 2.4</b> Pengkabelan <i>Coin Acceptor</i> CH-926 .....	11
<b>Gambar 2.5</b> Bagian-bagian <i>Coin Acceptor</i> CH-926 .....	11
<b>Gambar 2.6</b> LCD ( <i>Liquid CrystalDisplay</i> ) .....	13
<b>Gambar 2.7</b> <i>Limit Switch</i> .....	14
<b>Gambar 2.8</b> <i>MicroSD</i> .....	17
<b>Gambar 2.9</b> <i>Real Time Clock</i> DS1307.....	19
<b>Gambar 3.1</b> Blok Diagram Alat Tugas Akhir .....	21
<b>Gambar 3.2</b> Rangkaian Mikrokontroler ATmega 328 .....	24
<b>Gambar 3.3</b> Mikrokontroler ATmega 328 .....	26
<b>Gambar 3.4</b> Rangkaian <i>DC Power Supply</i> .....	27
<b>Gambar 3.5</b> <i>DC Power Supply</i> .....	28
<b>Gambar 3.6</b> Rangkaian <i>Driver Relay</i> .....	28
<b>Gambar 3.7</b> <i>Driver Relay</i> .....	29
<b>Gambar 3.8</b> Rangkaian <i>Connector</i> IC-MAX232 dengan DB9 .....	32
<b>Gambar 3.9</b> <i>Connector</i> IC-MAX232 dengan DB9 .....	33
<b>Gambar 3.10</b> Flowchart Pembayaran Sewa Loker .....	34
<b>Gambar 3.11</b> Program Pembayaran Sewa Loker .....	35
<b>Gambar 3.12</b> Flowchart Pembayaran Tarif Penggunaan .....	36
<b>Gambar 3.13</b> Program Pembayaran Tarif Penggunaan.....	37
<b>Gambar 3.14</b> Flowchart <i>Input Sensor Coin Acceptor</i> .....	38
<b>Gambar 3.15</b> Program <i>Input Sensor Coin Acceptor</i> .....	39
<b>Gambar 3.16</b> Flowchart Akses Waktu Modul <i>Real Time Clock</i> .....	40
<b>Gambar 3.17</b> Program Akses Waktu Modul <i>Real Time Clock</i> .....	41
<b>Gambar 3.18</b> Flowchart <i>Timer Internal</i> .....	42
<b>Gambar 3.19</b> Program <i>Timer Internal</i> .....	43
<b>Gambar 3.20</b> Flowchart Pengontrol <i>Driver Relay Solenoid DoorLock</i> ..	44
<b>Gambar 3.21</b> Program Pengontrol <i>Driver Relay Solenoid DoorLock</i> ..	45
<b>Gambar 3.22</b> Flowchart Serial Print Melalui RS232 .....	45
<b>Gambar 3.23</b> Bentuk Loker Tampak Depan .....	46
<b>Gambar 3.24</b> Bentuk Loker Tampak Belakang .....	47

<b>Gambar 3.25</b>	Bentuk Loker Tampak Samping .....	47
<b>Gambar 3.26</b>	Loker Tampak Depan .....	48
<b>Gambar 3.27</b>	Pemasangan <i>Solenoid Doorlock</i> .....	48
<b>Gambar 3.28</b>	Penempatan <i>Coin Acceptor</i> dan Rangkaian elektronika .....	49
<b>Gambar 3.29</b>	Loker Tampak Samping .....	49
<b>Gambar 3.30</b>	Loker Tampak Belakang .....	50
<b>Gambar 4.1</b>	Tampilan Kebutuhan Jumlah <i>Coin</i> .....	59
<b>Gambar 4.2</b>	Tampilan Ketika <i>Coin</i> yang Ditambahkan Sesuai .....	60
<b>Gambar 4.3</b>	Tampilan Ketika <i>Coin</i> yang Ditambahkan Tidak Sesuai .....	60
<b>Gambar 4.4</b>	Tampilan <i>Timer</i> dan Kebutuhan <i>Coin</i> .....	61
<b>Gambar 4.5</b>	Tampilan Perintah Memasukkan <i>Password</i> .....	62
<b>Gambar 4.6</b>	Tampilan Ketika <i>Password</i> Salah.....	62
<b>Gambar 4.7</b>	Tampilan Ketika <i>Password</i> Benar .....	62

## DAFTAR TABEL

	HALAMAN
<b>Tabel 4.1</b>	Hasil Pengukuran Tegangan <i>Output DC Power Supply</i> Tanpa Beban .....51
<b>Tabel 4.2</b>	Hasil Pengukuran Tegangan <i>Output DC Power Supply</i> Dengan Beban.....51
<b>Tabel 4.3</b>	Hasil Pengukuran Tegangan <i>Output</i> pin Mikrokontroler ATmega 328 .....52
<b>Tabel 4.4</b>	Hasil Pengukuran Tegangan <i>Input</i> yang diperlukan <i>Solenoid Doorlock</i> .....53
<b>Tabel 4.5</b>	Hasil Pengujian <i>Timer Internal</i> Mikrokontroler ATmega 328 .....54
<b>Tabel 4.6</b>	Hasil Modul RTC( <i>Real Time Clock</i> ) DS 1307 .....54
<b>Tabel 4.7</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.100 Perak .....56
<b>Tabel 4.8</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.100 Kuning .....56
<b>Tabel 4.9</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.200 Perak .....56
<b>Tabel 4.10</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.500 Perak .....57
<b>Tabel 4.11</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.500 Kuning .....57
<b>Tabel 4.12</b>	Hasil Pengujian <i>Coin Acceptor</i> Dengan <i>Coin</i> Rp.1000 Perak .....57
<b>Tabel 4.13</b>	Hasil Pengujian Tampilan <i>Serial Monitor</i> .....58
<b>Tabel 4.14</b>	Hasil Pengujian Kinerja Alat Keseluruhan .....63

-----Halaman ini sengaja dikosongkan-----

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Tempat umum adalah suatu tempat yang umumnya terdapat banyak orang yang berkumpul untuk melakukan suatu kegiatan baik secara sementara maupun secara terus menerus dan baik membayar maupun tidak membayar.[1]

Namun karena banyaknya pengunjung yang datang maka tak luput dari adanya tindak kriminalitas. Dimana para pelaku tindak kriminalitas akan mengincar terhadap barang-barang bawaan para pengunjung. Untuk mengantisipasi dan meminimalisir tindak kriminalitas tersebut maka pada tempat-tempat umum disediakan tempat penitipan barang.

Salah satu contoh tempat penitipan barang yang dapat diletakkan pada tempat umum adalah loker yang biasanya diawasi oleh seseorang yang telah ditugaskan. Namun loker yang ada pada penitipan barang di tempat umum saat ini masih menggunakan pengunci manual, dimana penguncinya menggunakan kunci berbentuk fisik.

Seiring dengan perkembangan teknologi yang semakin maju dan semakin canggih, maka dapat diciptakan loker penitipan barang yang memiliki pengunci otomatis. Dimana loker tersebut dapat menggunakan fitur *password* sebagai pengunci yang dapat diatur oleh para pengguna sesuka hati, dan pengguna dapat secara langsung membayar tarif penggunaan loker tersebut. Untuk itu pada loker penitipan tersebut dapat ditambahkan perangkat pembayaran. Sehingga tempat penitipan barang yang berupa loker pada tempat umum memiliki keamanan yang lebih baik dan tidak lagi memerlukan seorang petugas untuk menjaga keamanan loker maupun sebagai tempat pengguna loker untuk melakukan proses pembayaran.

### **1.2 Permasalahan**

Adapun permasalahan yang mendasari pembuatan Tugas Akhir ini adalah loker yang digunakan di tempat-tempat umum pada saat ini masih menggunakan sistem penguncian manual dengan sebuah kunci berbentuk fisik. Namun penggunaan kunci tersebut dapat menimbulkan masalah, salah satunya ketika kunci loker yang digunakan oleh pengunjung hilang yang menyebabkan loker tersebut tidak dapat dibuka. Sedangkan untuk loker penitipan yang memberlakukan tarif penitipan

masih diperlukan petugas sebagai penjaga sekaligus tempat pengguna untuk membayar biaya penggunaan loker.

### 1.3 Tujuan

Tujuan utama dari Tugas Akhir ini adalah menciptakan sebuah loker dengan pengunci otomatis yang aman dengan menggunakan *password* sebagai tempat penyimpanan umum yang dilengkapi dengan alat pembayaran otomatis. disertai penyimpanan data penitipan yang dapat diakses oleh pengguna.

### 1.4 Batasan Masalah

Batasan masalah dalam pembuatan prototype sistem kunci otomatis loker dapat diuraikan sebagai berikut:

1. Setiap loker memiliki satu sistem pembayaran
2. Alat pembayaran berupa uang logam Rp.500
3. Loker dapat diakses dan memasukkan *password* setelah melalui tahap pembayaran.
4. Loker dapat dikunci dan dibuka dengan *password*.
5. Satu *password* hanya berlaku untuk satu kali menggunakan loker
6. Data penitipan disimpan pada *MicroSD*
7. Komunikasi serial melalui RS232

### 1.5 Metodologi Penelitian

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu, tahap persiapan, perancangan dan pembuatan alat, tahap pengujian dan analisa, dan yang terakhir adalah penyusunan laporan berupa buku Tugas Akhir.

Pada tahap persiapan akan dilakukan studi literatur mengenai dasar teori serta mempelajari bahan dan komponen yang dibutuhkan dalam pembuatan Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran, seperti sensor *Coin* yang akan digunakan sebagai pembayaran, ATmega 328 sebagai pengontrol utama, *Driver Relay* untuk mengendalikan *solenoid door lock*, LCD (*liquid crystal display*) 20x4 untuk tampilan menu , *keypad* 3x4 Sebagai fasilitas untuk memasukkan *password*, *MicroSD* sebagai penyimpanan data penggunaan, RS232 sebagai komunikasi serial antara loker dan komputer , dan lain sebagainya.

Tahap perencanaan dan pembuatan alat akan dilakukan perancangan *hardware* dan *software* sesuai data yang didapat dari literatur. Untuk perancangan *hardware* didesain sebuah blok diagram

dengan menggunakan komponen dan cara pembuatannya berdasarkan dari studi literatur. Pada tahap ini program pada ATmega 328 dengan menggunakan pemrograman arduino.

Kemudian dilakukan tahap pengujian alat, pengambilan data alat, kemudian menganalisa kesalahan atau kegagalan pada alat dan mengatasi permasalahan tersebut. Dari hasil analisis, akan ditarik kesimpulan dari pengujian alat yang telah dilakukan. Tahap akhir metode yang dilakukan adalah penyusunan laporan untuk buku Tugas Akhir.

## **1.6 Sistematika Laporan**

Pembahasan Tugas Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

### **Bab I Pendahuluan**

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

### **Bab II Teori Dasar**

Bab ini menjelaskan tentang tinjauan pustaka dan *hardware* yang digunakan, yaitu *solenoid doorlock*, *Coin acceptor* CH-926, Mikrokontroler ATmega 328, *Driver Relay*, *keypad* 3x4 , LCD (*liquid crystal display*) 20X4, modul RTC (*Real Time Clock*) , *limith switch*, RS232 , MicroSD Sedangkan *software* yang digunakan adalah arduino.

### **Bab III Perancangan dan Pembuatan Alat**

Bab ini membahas desain dan perancangan sistem kunci otomatis dan sistem pembayaran pada loker baik *hardware* maupun *software* berdasarkan teori dasar pada Bab II.

### **Bab IV Hasil Pengujian dan Pengukuran Alat**

Bab ini memuat data – data hasil dari pengujian dan pengukuran alat pada *hardware* dan *software*.

## **Bab V Penutup**

Bab ini berisi kesimpulan dari proses-proses yang dilalui selama pembuatan Tugas Akhir ini dan saran saran untuk pengembangan alat selanjutnya.

### **1.7 Relevansi**

- a. Untuk menciptakan sistem pengunci otomatis pada loker menggunakan *password*
- b. Untuk menciptakan sistem pembayaran pada loker dengan *Coin acceptor*
- c. Untuk menciptakan alat penitipan barang yang aman, berupa loker yang dapat digunakan oleh umum



## **BAB II**

### **TEORI DASAR**

Bab ini membahas tentang teori dasar komponen dan alat yang digunakan dalam membuat alat Tugas Akhir Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran. Dimana teori dasar didapat dari berbagai referensi yang telah dicantumkan pada daftar pustaka. Uraian teori terdiri dari *hardware* dan *software* yang digunakan, antara lain: Mikrokontroler Atmega 328, *DC Power Supply*, *Solenoid Door Lock*, *Coin Acceptor CH-926*, *LCD (Liquid Crystal Display) 20x4*, *Keypad 3x4*, Modul RTC (*Real Time Clock*), *MicroSD*, *RS232*, *MicroSD*, *Driver Relay*, *Limit Switch*, Pemrograman Arduino.

#### **2.1 Mikrokontroler ATmega 328**

Mikrokontroler merupakan sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronika. Berbeda dengan mikroprosesor, mikrokontroler memiliki prosesor, memori, dan I/O yang terintegrasi menjadi satu kesatuan dalam satu chip. Sedangkan mikroprosesor tidak terintegrasi dan membutuhkan tambahan chip memori yang terpasang secara eksternal. Mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data. Contohnya pada mikrokontroler yang digunakan dalam Tugas Akhir ini adalah mikrokontroler ATmega 328. ATmega 328, seperti pada Gambar 2.1 merupakan mikrokontroler keluaran AVR 8 bit. ATmega 328 dapat di bootloader dengan arduino sehingga dapat dikoneksikan dengan bahasa pemrograman bawaan arduino. ATmega 328 memiliki 3 buah PORT utama yaitu PORT B, PORT C, dan PORT D dengan total pin *Input/output* sebanyak 23 pin. PORT tersebut dapat difungsikan sebagai *Input/output* digital atau difungsikan sebagai peripheral lainnya.

Adapun fungsi-fungsi dari pin pada mikrokontroler ATmega 328 adalah sebagai berikut:[3]

##### **1. PORT B**

Port B merupakan jalur data 8 bit yang dapat difungsikan sebagai *Input* atau *Output*. Selain itu Port B juga dapat digunakan untuk fungsi Alternatif sebagai berikut ini:

- a. ICP1 (PB0), berfungsi sebagai *Timer Counter 1 Input capture* pin.
- b. OC1A (PB1), OC1B (PB2), dan OC2 (PB3) merupakan pin yang dapat difungsikan sebagai keluaran PWM (*Pulse Width Modulation*).
- c. MOSI (PB3), MISO (PB4), SCK (PB5), SS (PB2) adalah jalur komunikasi SPI selain itu berfungsi sebagai jalur pemrograman serial (ISP).
- d. TOSC1 (PB6) dan TOSC2 (PB7) merupakan sumber clock external untuk timer.
- e. XTAL1 (PB6) dan XTAL2 (PB7) merupakan sumber clock utama Mikrokontroler.

## 2. PORT C

Port C merupakan jalur data 7 bit yang dapat difungsikan sebagai *Input* atau *ouput* digital. Fungsi alternatif port C antara lain sebagai berikut.

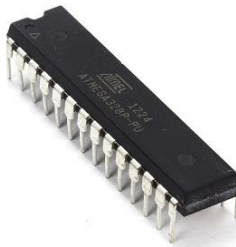
- a. ADC6 channel (PC0, PC1, PC2, PC3, PC4, PC5) dengan resolusi yang besarnya 10 bit. ADC dapat digunakan untuk mengubah *Input* yang berupa tegangan analog menjadi data digital.
- b. 12C (SDA dan SDL) merupakan salah satu fitur yang terdapat pada PORT C. 12C digunakan untuk komunikasi dengan sensor atau device lain yang memiliki komunikasi data jenis 12C seperti pada sensor kompas, accelerometer nunchuck.

## 3. PORT D

Port D merupakan jalur data 8 bit yang masing-masing pinnya dapat difungsikan sebagai *Input* atau *ouput*. Sama seperti port B dan port C, port D juga memiliki fungsi alternatif dibawah ini.

- a. USART (TXD dan RXD) merupakan jalur data komunikasi serial dengan level sinyal TTL. Pin TXD ,berfungsi untuk mengirimka data serial, sedangkan RXD yaitu pin yang berfungsi sebagai menerima data serial.
- b. Interrupt (INT0 dan INT1) merupakan pin dengan fungsi khusus sebagai interupsi *hardware*. Interupsi biasanya digunakan sebagi selaan dari program, misalnya pada saat program yang berjalan kemudian terjadi interupsi

- hardware /software* maka program utama akan berhenti dan akan menjalankan program interupsi.
- c. XCK dapat difungsikan sebagai sumber clock external untuk USART, namun juga dapat memanfaatkan clock dari CPU, sehingga tidak perlu menggunakan external clock.
  - d. T0 dan T1 berfungsi sebagai masukan counter external untuk timer 1 dan timer 0.
  - e. AIN0 dan AIN1 keduanya merupakan masukan *Input* untuk analog comparator.



**Gambar 2.1** Mikrokontroler ATmega 328

## **2.2 DC Power Supply**

Arus Listrik yang kita gunakan di rumah, kantor dan pabrik pada umumnya adalah dibangkitkan, dikirim dan didistribusikan ke tempat masing-masing dalam bentuk Arus Bolak-balik atau arus AC (Alternating Current). Hal ini dikarenakan pembangkitan dan pendistribusian arus Listrik melalui bentuk arus bolak-balik (AC) merupakan cara yang paling ekonomis dibandingkan dalam bentuk arus searah atau arus DC (Direct Current). Akan tetapi, peralatan elektronika yang kita gunakan sekarang ini sebagian besar membutuhkan arus DC dengan tegangan yang lebih rendah untuk pengoperasiannya. Oleh karena itu, hampir setiap peralatan Elektronika memiliki sebuah rangkaian yang berfungsi untuk melakukan konversi arus listrik dari arus AC menjadi arus DC dan juga untuk menyediakan tegangan yang sesuai dengan rangkaian Elektronika-nya[4]

Rangkaian yang mengubah arus listrik AC menjadi DC ini disebut dengan DC Power Supply atau dalam bahasa Indonesia disebut dengan Catu daya DC. DC Power Supply atau Catu Daya ini juga sering dikenal dengan nama “Adaptor”. Sebuah DC Power Supply atau Adaptor pada

dasarnya memiliki 4 bagian utama agar dapat menghasilkan arus DC yang stabil. Keempat bagian utama tersebut diantaranya adalah Transformer, Rectifier, Filter dan Voltage Regulator[4].

#### 1. Transformator (Transformer / Trafo)

Transformator (Transformer) atau disingkat dengan Trafo yang digunakan untuk DC Power Supply adalah Transformer jenis Step-down yang berfungsi untuk menurunkan tegangan listrik sesuai dengan kebutuhan komponen Elektronika yang terdapat pada rangkaian adaptor (DC Power Supply). Transformator bekerja berdasarkan prinsip Induksi elektromagnetik yang terdiri dari 2 bagian utama yang berbentuk lilitan yaitu lilitan Primer dan lilitan Sekunder. Lilitan Primer merupakan *Input* dari pada Transformator

sedangkan *Output*-nya adalah pada lilitan sekunder. Meskipun tegangan telah diturunkan, *Output* dari Transformator masih berbentuk arus bolak-balik (arus AC) yang harus diproses selanjutnya

#### 2. Penyearah Gelombang (Rectifier)

Rectifier atau penyearah gelombang adalah rangkaian Elektronika dalam Power Supply (catu daya) yang berfungsi untuk mengubah gelombang AC menjadi gelombang DC setelah tegangannya diturunkan oleh Transformator Step down. Rangkaian Rectifier biasanya terdiri dari komponen Dioda. Terdapat 2 jenis rangkaian Rectifier dalam Power Supply yaitu “Half Wave Rectifier” yang hanya terdiri dari 1 komponen Dioda dan “Full Wave Rectifier” yang terdiri dari 2 atau 4 komponen dioda.

#### 3. Penyaring (Filter)

Dalam rangkaian DC Power Supply, filter digunakan untuk meratakan sinyal arus yang keluar dari Rectifier. Filter ini biasanya terdiri dari komponen Kapasitor (Kondensator) yang berjenis Elektrolit atau ELCO(Electrolyte Capacitor

#### 4. Pengatur Tegangan (Voltage Regulator)

Untuk menghasilkan Tegangan dan Arus DC (arus searah) yang tetap dan stabil, diperlukan Vage Regulator yang berfungsi untuk mengatur tegangan sehingga tegangan *Output* tidak dipengaruhi oleh suhu, arus beban dan juga tegangan *Input* yang berasal *Output* Filter. Vage Regulator pada umumnya terdiri dari Dioda Zener, Transistor atau IC (Integrated Circuit). Pada DC Power Supply yang canggih, biasanya Vage Regulator juga dilengkapi dengan Short Circuit Protection (perlindungan atas hubung singkat), Current Limiting

(Pembatas Arus) ataupun OverVage Protection (perlindungan atas kelebihan tegangan).

### 2.3 *Solenoid Doorlock*

*Solenoid Doorlock* adalah sebuah pengunci pintu yang mengaplikasikan sistem *solenoid*. *Solenoid* adalah sebuah kumparan eleKtromagnet yang dirancang secara khusus. Cara kerja *solenoid* ini adalah pada saat arus mengalir melalui kawat pada sistem *solenoid*, disekitar kawat tersebut akan menghasilkan medan magnet. Sistem *solenoid* menggunakan kumparan yang terdiri dari gulungan kawat yang diperbanyak, sehingga medan magnet yang dihasilkan akan lebih besar dan mengalir disekitar kumparan kawat tersebut. Pada kumparan tersebut nantinya akan dipasang.[5]

Sebuah pegas yang nantinya jika medan magnetnya terbentuk pegas tersebut akan tertarik oleh magnet tersebut. Pada *Solenoid Door lock* apabila dialiri arus maka kunci terbuka dan apabila arus listrik diputus maka pegas akan meregang kembali karena medan magnet hilang. bentuk *Solenoid Doorlock* dapat dilihat pada Gambar 2.2.



**Gambar 2.2** *Solenoid Doorlock*

### 2.4 *Coin Acceptor CH-926*

*Coin Acceptor* CH-926 merupakan sensor *Coin* yang dapat digunakan untuk mendeteksi berbagai jenis *Coin*. Banyaknya jenis *Coin* yang dapat dideteksi oleh *Coin acceptor* tergantung dari jenis *Coin acceptor* yang digunakan. Untuk *Coin acceptor* CH-926 seperti terlihat pada Gambar 2.3 banyaknya jenis *Coin* yang dapat dideteksi adalah 6 jenis *Coin*. *Coin acceptor* bekerja mengenali jenis *Coin* berdasarkan pada bahan, berat dan diameter dari *Coin* tersebut. CH-926, seperti

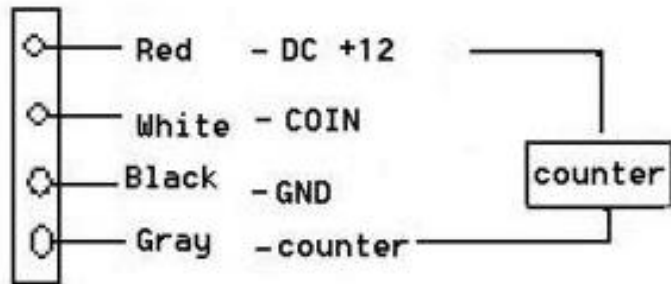
Gambar 2.3 sangat stabil dan akurat sekalipun ada perubahan pada lingkungan sekitar, seperti temperature dan kelembaban Untuk meningkatkan akurasi maka sangat disarankan agar setiap chanel dipakai untuk mengenali satu macam Coin saja[6]. Dalam penggunaannya maka perlu mengetahui spesifikasi dari CH-926, diantaranya:[7]

1. Diameter *Coin* yang diperbolehkan : 15mm~32mm
2. Ketebalan *Coin* yang diperbolehkan :1.2mm~3.8mm
3. Tegangan kerja : DC +12V  $\pm$ 10%
4. Konsumsi arus : 65mA  $\pm$ 5%
5. Akurasi pendeteksian : 99.5%
6. Sinyal keluaran : pulse
7. Kecepatan deteksi :  $\leq 0.6s$  (tergantung jenis *Coin*)

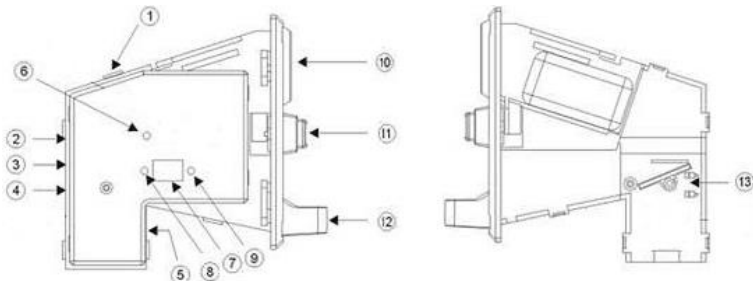
Sebelum menggunakan sensor *Coin* ini, perlu mengetahui bagian – bagiannya untuk mengetahui *Input* dan *ouput* dari sensor *Coin* ini . Untuk sistem pengkabelan *Coin acceptor* CH-926 dapat dilihat pada Gambar 2.4. Sedangkan Untuk bagian-bagian *Coin acceptor* CH-926 dapat dilihat pada Gambar 2.5.



**Gambar 2.3** *Coin Acceptor* CH-926



**Gambar 2.4** Pengkabelan *Coin Acceptor* CH-926



**Gambar 2.5** Bagian-bagian *Coin Acceptor* CH-926

Bagian-bagian dari *Coin acceptor* CH-926 adalah sebagai berikut:

1. LED Indikator
2. Konektor 4 pin
3. Pemilih – jenis sunyal kemuaran NO atau NC.
4. Pemilih lebar pulsa sinyal keluaran – fast: 20ms , medium: 50ms , slow: 100ms
5. Soket untuk – Electromagnetic Valve DC 12V
6. Tombol “Setup”
7. Display seven segmen
8. Tombol “Add” – untuk menambah nilai setting
9. Tombol “Minus” – untuk mengurangi nilai setting
10. Slot *Coin*, untuk memasukkan *Coin*
11. Tombol untuk mengeluarkan *Coin* yang tertahan
12. Keluaran *Coin* yang tertahan atau tertolak

13. Letak solenoid yang akan terbuka jika *Coin* diterima, sehingga *Coin* akan jatuh ke bawah.

## 2.5 LCD (*Liquid Crystal Display*) 20x4

LCD(*Liquid Crystal Display*) adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis display elektronik yang dirancang dengan teknologi CMOS logic yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit.[8]

LCD, seperti Gambar 2.6 terdiri dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. LCD 20x4 artinya memiliki 20 kolom x 4 baris. Tampilan dari LCD dilengkapi dengan *backlight* yang bermacam-macam warnanya, antara lain hijau, merah, putih, dan biru. Kaki atau jalur *Input* dan kontrol dalam suatu LCD (*Liquid Cristal Display*) antara lain[8] :

1. Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Cristal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti Mikrokontroler dengan lebar data 8 bit.
2. Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika low menunjukkan yang masuk adalah perintah, sedangkan logika high menunjukkan data.
3. Pin R/W (*Read Write*) berfungsi sebagai instruksi pada modul jika low tulis data, sedangkan high baca data.
4. Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.
5. PinVLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 K $\Omega$ , jika tidak digunakan dihubungkan ke ground, sedangkan tegangan catu daya ke LCD sebesar 5V.

Karakteristik dari LCD terdapat 80 x 8 bit *display* RAM dengan kapasitas maksimal 80 karakter dan memiliki kemampuan penulisan



dengan 8 bit maupun 4 bit. LCD dibangun dengan osilator local dengan catu daya tegangan sebesar 5V dimana saat tegangan dihidupkan reset akan bekerja secara otomatis. Rentang suhu kerja LCD ini antara 0 C sampai 55 C.



**Gambar 2.6** LCD (*Liquid Crystal Display*)

## **2.6 Limit Switch**

*Limit switch* merupakan jenis saklar yang dilengkapi dengan katup yang berfungsi menggantikan tombol. Prinsip kerja *limit switch* sama seperti saklar Push ON yaitu hanya akan menghubungkan pada saat katupnya ditekan pada batas penekanan tertentu yang telah ditentukan dan akan memutus saat katup tidak ditekan. *Limit switch* termasuk dalam kategori sensor mekanis yaitu sensor yang akan memberikan perubahan elektrik saat terjadi perubahan mekanik pada sensor tersebut. Penerapan dari *limit switch* adalah sebagai sensor posisi suatu benda (objek) yang bergerak[13]

Prinsip kerja *limit switch* seperti yang terlihat pada Gambar 2.7 diaktifkan dengan penekanan pada tombolnya pada batas atau daerah yang telah ditentukan sebelumnya sehingga terjadi pemutusan atau penghubungan rangkaian dari rangkaian tersebut. Limit switch memiliki 2 kontak yaitu NO (*Normally Open*) dan kontak NC (*Normally Close*) dimana salah satu kontak akan aktif jika tombolnya tertekan.[13]



**Gambar 2.7** *Limit Switch*

## 2.7 *Driver Relay*

Rangkaian *Driver Relay* memiliki arti sebagai rangkaian elektronika yang biasanya digunakan untuk mengendalikan serta pengoperasian sesuatu dari jarak jauh atau semacam remote. Tentunya rangkaian ini bisa mempermudah dan juga mempercepat pekerjaan yang memang kadang membutuhkan rangkaian dari relay ini. Dengan menggunakan rangkaian relay tersebut, dapat melakukan kontrol dan juga mengoperasikan perangkat elektronik dari jarak jauh .[9]

Bagian utama dari *Driver Relay* adalah yang merupakan salah satu komponen yang didalamnya terdiri dari sebuah kumparan berinti besi yang akan menghasilkan elektromagnet ketika kumparannya dialiri oleh arus listrik. Elektromagnet ini kemudian menarik mekanisme kontak yang akan menghubungkan kontak Normally-Open (NO) dan membuka kontak Normally-Closed (NC). Sedikit menjelaskan, kata Normally disini berarti relay dalam keadaan non-aktif atau non-energized, atau gambangnya kumparan relay tidak dialiri arus. Jadi kontak Normally-Open (NO) adalah kontak yang pada saat Normal tidak terhubung, dan kontak Normally-Closed (NC) adalah kontak yang pada saat Normal terhubung.[10]

## 2.8 *Komunikasi Serial RS232*

Untuk mendapatkan keserba-cocokan (compatibility) dari beberapa peralatan komunikasi data dari berbagai pabrik, diciptakanlah standar antar-muka (*interfacing*) yang dinamakan RS232. Standar ini dipublikasikan oleh EIA (*Electronics Industries Association*) pada 1960. Pada 1963 standar tersebut dimodifikasi dengan nama RS232A. RS232B dan RS232C ditetapkan pada tahun masing-masing 1965 dan 1969. Pada

RS232, 1s (high) direpresentasikan dengan tegangan -3 s/d -25V, dan 0s (low) direpresentasikan sebagai +3 s/d +25V. Sedang diantara -3 dan +3V dianggap sebagai status mengambang dan tidak dianggap.[12]

Perangkat yang menggunakan kabel serial untuk komunikasinya dibagi ke dalam dua kategori. Yaitu DCE (*Data Communications Equipment*) dan DTE (*Data Terminal Equipment*). *Data Communications Equipment* adalah perangkat seperti modem, TA adapter, plotter dan lain-lain, sedangkan *Data Terminal Equipment* adalah komputer anda atau Terminal. Untuk menjamin terjadinya sebuah transfer data yang cepat dan Realible antara 2 peralatan, lalu lintas data harus dikoordinasi dengan baik. Tidak seperti printer yang selalu mencetak setiap karakter yang diterimanya. Namun dalam komunikasi serial, bisa saja peralatan tidak memiliki lagi tampungan data yang diterimanya. Sehingga dia harus memberitahukan PC untuk tidak lagi mengirim data. Hingga modem selesai mengerjakan semua tugasnya. Dan kembali memberitahukan PC[12]

untuk kembali mengirim data berikutnya setelah modem siap

Keterangan mengenai fungsi saluran RS232 pada konektor DB9 adalah sebagai berikut:

1. Pin *Carrier Detect*(CD) Saat perangkat mendeteksi suatu carier dari perangkat lain, maka sinyalsinyal ini akan diaktifkan
2. Pin *Receive Data*(RD) Berfungsi sebagai penerima data serial (RDX)
3. Pin *Transmite Data* (TD) Berfungsi sebgai pengurum data serial (TDX)
4. Pin *Data Terminal Ready* (DTR) Berfungsi untuk memberitahu Bahwa UART siap melakukan pertukaran data
5. Pin *Signal Ground* (SG) Berfungsi untuk dihubungkan ke ground
6. Pin *Data Set Ready* (DSR) Berfungsi memberitahukan UART bahwa perangkat siap melakukan pertukaran data
7. Pin *Request To Send* (RTS) Berfungsi sebgai sinyal untuk menginformasikan perangkat bahwa UART siap melakukan pertukaran data
8. Pin *Clear To Send* (CTS) Berfungsi untuk memberitahukan bahwaperangkat siap melakukan pertukaran data
9. Pin *Ring Indikator* (RI) akan aktif ketika ada sinyal yang masuk

Kecepatan tranfer data pada komunikasi data serial diukur dalam satuan BPS (bits persecond). Sebutan terkenal lainnya adalah baud rate.

Namun Baud dan bps tidak serta merta adalah sama. Hal ini mengacu kepada fakta bahwa baud rate adalah terminology modem dan diartikan sebagai perubahan signal dalam satuan bit signal setiap detik. Sedangkan data tranfer rate penamaannya mengacu pada jumlah bit dari byte data yang ditransfer setiap detik[12].

Sementara itu kecepatan transfer data (data transfer rate) pada komputer tergantung pada jenis komunikasi yang diberlakukan atasnya. Seperti contoh, komputer PC-IBM model kuno dapat mentransfer data mulai dari 100 s/d 9600 bps. Namun pada saat sekarang kecepatan komunikasi serial menjadi sangat pesat. 56.000 bps kemudian menjadi standar kecepatan pada modem. Namun para perancang komputer sepakat untuk membatasi kecepatan pada komunikasi serial Asynchronous hanya setinggi 100.000 bps. Untuk kecepatan yang lebih tinggi mode Synchronous kemudian menjadi pilihan[12].

## 2.9 *MicroSD*

*MicroSD* adalah kartu memori non-volatile yang dikembangkan oleh *SD Card Association* yang digunakan dalam perangkat *portable* [13] seperti pada Gambar 2.8 . Saat ini, teknologi *microSD* sudah digunakan oleh lebih dari 400 merek produk serta dianggap sebagai standar industri de-facto.

Keluarga *microSD* yang lain terbagi menjadi SDSC yang kapasitas maksimum resminya sekitar 2GB, meskipun beberapa ada yang sampai 4GB. SDHC (*High Capacity*) memiliki kapasitas dari 4GB sampai 32GB. Dan SDXC (*Extended Capacity*) kapasitasnya di atas 32GB hingga maksimum 2TB[13]

Dari sudut pandang perangkat, semua kartu ini termasuk kedalam keluarga SD. SD adapter memungkinkan konversi fisik kartu SD yang lebih kecil untuk bekerja di slot fisik yang lebih besar dan pada dasarnya ini adalah alat pasif yang menghubungkan pin dari *microSD* yang kecil ke pin adaptor *microSD* yang lebih besar[13].

SD mempunyai bentuk fisik yang sama maka sering menyebabkan kebingungan di kalangan konsumen. Contohnya, *MicroSD*, *MicroSDHC*, dan *MicroSDXC* ukuran fisiknya sama tetapi kapabilitasnya berbeda. Protokol komunikasi untuk SDHC/SDXC/SDIO sedikit berbeda dengan *MicroSD* yang sudah mapan karena biasanyahost device keluaran lama tidak bisa mengenali kartu keluaran baru. kebanyakan masalah mengenai inkompatibilitas ini dapat diselesaikan dengan *firmware update*. [13]



**Gambar 2.8** *MicroSD*

### **2.10 Modul RTC (*Real Time Clock*) DS 1307**

*Real Time Clock* merupakan suatu chip (IC) yang memiliki fungsi sebagai penyimpan waktu dan tanggal. DS1307 merupakan *Real-time clock* (RTC) yang dapat menyimpan data-data detik, menit, jam, tanggal, bulan, hari dalam seminggu, dan tahun valid hingga 2100. 56-byte, *battery-backed*, RAM *nonvolatile* (NV) RAM untuk penyimpanan. DS1307 seperti yang terlihat pada Gambar 2.9 merupakan *Real-time clock* (RTC) dengan jalur data parallel yang memiliki Antarmuka serial *Two-wire* (I2C), Sinyal luaran gelombang-kotak terprogram (*Programmable squarewave*), Deteksi otomatis kegagalan-daya (*power-fail*) dan rangkaian *switch*, Konsumsi daya kurang dari 500nA menggunakan mode baterai cadangan dengan operasional osilator. Tersedia fitur industri dengan ketahanan suhu: -40°C hingga +85°C. Tersedia dalam kemasan 8-pin DIP atau SOIC.[14]

Fungsi pin pada DS 1307 :

1. Pin X1

Merupakan pin yang digunakan untuk dihubungkan dengan kristal sebagai pembangkit clock.

2. Pin X2

Berfungsi sebagai keluaran/*Output* dari crystal yang digunakan. Terhubung juga dengan X1.

3. PinVBAT

Merupakan *backup supply* untuk RTC DS1307 dalam menjalankan fungsi waktu dan tanggal. Besarnya adalah 3V dengan menggunakan jenis *Lithium Cell* atau sumber energi lain. Jika pin ini tidak digunakan maka harus terhubung dengan *Ground*. Sumber tegangan dengan 48mAH atau lebih besar dapat digunakan sebagai

cadangan energi sampai lebih dari 10 tahun, namun dengan persyaratan untuk pengoperasian dalam suhu 25°C.

4. GND

*Ground* (GND) merupakan sebuah titik referensi umum atau tegangan potensial yang sama dengan “tegangan nol”. *Ground* bersifat relatif, karena dapat memilih titik dimana saja dalam sirkuit untuk dijadikan *ground* untuk mereferensi semua tegangan dalam rangkaian. *Ground* berfungsi untuk menetralkan cacat (noise) yang disebabkan baik oleh daya yang kurang baik, ataupun kualitas komponen yang tidak standar. Sistem grounding pada peralatan kelistrikan dan elektronika adalah untuk memberikan perlindungan pada seluruh sistem.

5. Pin SDA

Berfungsi sebagai masukan/keluaran (I/O) untuk I2C *serial interface*. Pin ini bersifat open drain, oleh sebab itu membutuhkan *eksternal pull up resistor*.

6. Pin SCL

Berfungsi sebagai clock untuk *Input* ke I2C dan digunakan untuk mensinkronisasi pergerakan data dalam *serial interface*. bersifat *open drain*, oleh sebab itu membutuhkan *eksternal pull up resistor*.

7. Pin SWQ/OUT

Sebagai *square wave* atau *Output Driver*. Jika diaktifkan, maka akan menjadi 4 frekuensi gelombang kotak yaitu 1Hz, 4kHz, 8kHz, 32kHz sifat dari pin ini sama dengan sifat pin SDA dan SCL sehingga membutuhkan *eksternal pull up resistor*. Dapat dioperasikan dengan VCC maupun dengan VBAT.

8. VCC

Merupakan sumber tegangan utama. Jika sumber tegangan terhubung dengan baik, maka pengaksesan data dan pembacaan data dapat dilakukan dengan baik. Namun jika *backup supply* terhubung juga dengan VCC, namun besar VCC di bawah VTP, maka pengaksesan data tidak dapat dilakukan.



**Gambar 2.9** Real Time Clock DS1307

## 2.11 Pemrograman Arduino

IDE Arduino adalah software yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari[15]:

1. Editor program, sebuah window yang memungkinkan pengguna menulis dan mengedit program dalam bahasa Processing.
2. Compiler, sebuah modul yang mengubah kode program (bahasa Processing) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa Processing. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya compiler diperlukan dalam hal ini.
3. Uploader, sebuah modul yang memuat kode biner dari komputer ke dalam memory didalam papan Arduino. Sebuah kode program Arduino umumnya disebut dengan istilah sketch. Kata “sketch” digunakan secara bergantian dengan “kode program” dimana keduanya memiliki arti yang sama.

Program Arduino sendiri menggunakan bahasa C. walaupun banyak sekali terdapat bahasa pemrograman tingkat tinggi (high level language) seperti pascal, basic, cobol, dan lainnya. Walaupun demikian, sebagian besar dari paraprogramer profesional masih tetap memilih bahasa C sebagai bahasa yang lebih unggul, berikut alasan-alasannya:[15]

- 1 Bahasa C merupakan bahasa yang powerful dan fleksibel yang telah terbukti dapat menyelesaikan program-program besar seperti pembuatan sistem operasi, pengolah Gambar (seperti pembuatan game) dan juga pembuatan kompilator bahasa pemrograman baru.
- 2 Bahasa C merupakan bahasa yang portable sehingga dapat dijalankan di beberapa sistem operasi yang berbeda. Sebagai contoh program yang kita tulis dalam sistem operasi windows

dapat kita kompilasi didalam sistem operasi linux dengan sedikit ataupun tanpa perubahan sama sekali.

3, Bahasa C merupakan bahasa yang sangat populer dan banyak digunakan oleh programmer berpengalaman sehingga kemungkinan besar library pemrograman telah banyak disediakan oleh pihak luar/lain dan dapat diperoleh dengan mudah.

4. Bahasa C merupakan bahasa yang bersifat modular, yaitu tersusun atas rutin-rutin tertentu yang dinamakan dengan fungsi (function) dan fungsi-fungsi tersebut dapat digunakan kembali untuk pembuatan program-program lainnya tanpa harus menulis ulang implementasinya.

5. Bahasa C merupakan bahasa tingkat menengah (middle level language) sehingga mudah untuk melakukan interface (pembuatan program antar muka) ke perangkat keras.

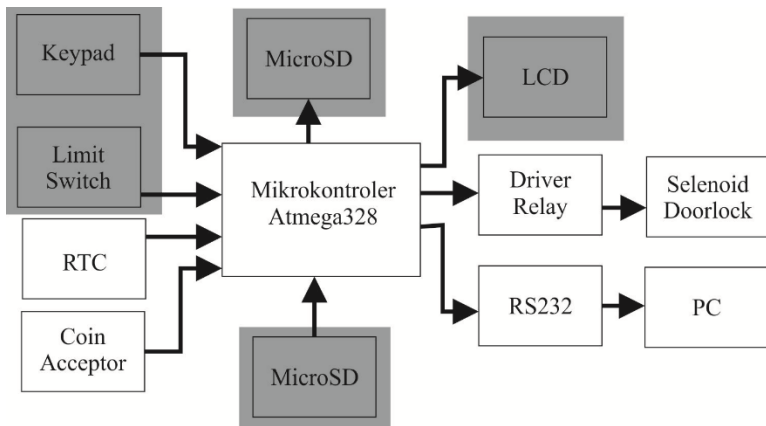
Struktur penulisan program dalam bahasa C harus memiliki fungsi utama, yang bernama `main()`. Fungsi inilah yang akan dipanggil pertama kali pada saat proses eksekusi program. Artinya apabila kita mempunyai fungsi lain selain fungsi utama, maka fungsi lain tersebut baru akan dipanggil pada saat digunakan. Oleh karena itu bahasa C merupakan bahasa prosedural yang menerapkan konsep runtutan (program dieksekusi per baris dari atas ke bawah secara berurutan), maka apabila kita menuliskan fungsi-fungsi lain tersebut dibawah fungsi utama, maka kita harus menuliskan bagian prototipe (prototype), hal ini dimaksudkan untuk mengenalkan terlebih dahulu kepada kompiler daftar fungsi yang akan digunakan di dalam program. Namun apabila kita menuliskan fungsi-fungsi lain tersebut diatas atau sebelum fungsi utama, maka kita tidak perlu lagi untuk menuliskan bagian prototipe diatas.[17]



### BAB III

## PERANCANGAN DAN PEMBUATAN ALAT

Pada bab ini akan dibahas mengenai tahapan yang dilakukan terhadap perancangan dan pembuatan Tugas Akhir loker penitipan otomatis dilengkapi dengan sistem pembayaran. Blok diagram alat Tugas Akhir pada Gambar 3.1 merupakan sistem kerja Tugas Akhir perancangan Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran. Sistem kerja secara keseluruhan di kontrol oleh Mikrokontroler ATmega 328 yang terintegrasi dengan rangkaian dan komponen – komponen pendukung. Adapun fungsi masing – masing komponen yang terdapat pada diagram perancangan alat Tugas Akhir, antara lain :



**Gambar 3.1** Blok Diagram Alat Tugas Akhir

1. Mikrokontroler ATmega 328  
Merupakan bagian terpenting pada sistem kerja yang berfungsi untuk mengolah *Input*, kemudian diproses untuk memberi perintah pada *ouput*
2. *Coin Acceptor*  
Fungsi dari *Coin Acceptor* sebagai tempat pembayaran loker dengan menggunakan uang Coin sebesar RP 500,-

3. *Keypad*  
Fungsi dari *keypad* sebagai *Input password*
4. LCD 20x4  
Berfungsi sebagai *interface password* dan perintah awal dari sistem
5. *Driver Relay*  
Digunakan untuk menggerakkan arus/tegangan seleoid
6. *Solenoid doorlock*  
Berfungsi untuk mengunci pintu loker
7. Modul *Real Time Clock*  
Merupakan modul yang berfungsi untuk menyediakan data waktu dan tanggal secara *Real Time* atau sesuai dengan waktu yang sebenarnya.
8. *Limit switch*  
Mendetaksi pintu loker dalam keadaan terbuka atau tertutup
9. RS232  
Sebagai komunikasi serial antara Arduino dengan komputer
10. MicroSD  
Sebagai tempat penyimpanan data penggunaan loker yang bisa digunakan oleh pengguna

Pada Blok Diagram diatas terdapat pembagian tugas dalam pengerjaan Tugas Akhir Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran. Dimana bagian yang dijelaskan pada buku ini adalah pada bagian berwarna putih, yaitu bagian *Coin Acceptor* dan Modul *Real Time Clock* sebagai *Input* pada Mikrokontroler ATmega 328, serta *Driver Relay* pengontrol *Solenoid Doorlock* dan RS232 sebagai *Output* dari Mikrokontroler ATmega 328. Sedangkan untuk bagian lainnya dikerjakan oleh Mahasiswa lain atas nama Raden Yochanan Adi S.

### 3.1 Perancangan *Hardware*

Perancangan *Hardware* dilakukan dengan merancang rangkaian elektronika dan rancangan mekanik. Perancangan *hardware* meliputi:

1. Rangkaian Mikrokontroler ATmega 328
2. Rangkaian *DC Power Supply*
3. Rangkaian *Driver Relay*
4. *Setting Coin Acceptor* CH-926
5. Perancangan *Connector* IC-MAX232

#### 3.1.1 Perancangan Mikrokontroler ATmega 328

Rangkaian Mikrokontroler ATmega 328 seperti pada Gambar 3.2 ini digunakan sebagai pengontrol atau pengendali dari seluruh proses yang dilakukan oleh alat. Dimana pada Mikrokontroler ATmega 328 memiliki 28 pin dengan 23 pin diantaranya terdiri dari 3 PORT yaitu PORT B yang memiliki 8 pin yaitu PORT PB0 sampai PB7, PORT C yang memiliki 7 pin yaitu PORT PC0 sampai PC6, Serta PORT D yang memiliki 8 pin antara PORT PD0 sampai PD7.

Adapun fungsinya antara lain sebagai pengunci otomatis *solenoid door lock* yang digerakkan oleh *Driver Relay*, membaca nilai pada *Coin acceptor*, mengatur tampilan dari *Input keypad* ke LCD 20x4, menerima masukan dari *Limith Switch* dan Proses penyimpanan data pada *MicroSD* serta komunikasi serial melalui RS232.

Tegangan supply VCC pada rangkaian Mikrokontroler ATmega 328 adalah 5VDC. Untuk mengisi program dengan menggunakan ICSP dengan menggunakan 4 sinyal MISO, MOSI, SCK dan RESET. Tegangan supply 5V dapat disupply dari USBASP programmer dengan mengambil daya dari port USB komputer.

Kristal resonator yang digunakan pada rangkaian di atas adalah dengan menggunakan 16 MHz. Resistor yang berfungsi sebagai pullup pada rangkaian di atas menggunakan 1k dan 10k. Kapasitor (C1 dan C2) pada rangkaian memiliki nilai sebesar 22 pF.



maka program yang ada pada pin INT) akan terlebih dahulu di jalankan.

3 Pengendali *Driver Relay*

*Driver Relay* yang digunakan memerlukan pemicu berupa sinyal digital dari *Output* pin Mikrokontroler ATmega 328, pada alat yang dibuat pin pemicu pada *Driver Relay* dihubungkan ke pin D16 , ketika pin D16 berlogika LOW maka *Driver Relay* tidak aktif , sedangkan ketika pin D16 berlogika HIGH maka *Driver Relay* aktif.

4. *Input Keypad*

*Keypad* yang digunakan pada alat ini adalah *keypad* 3x4 , *keypad* ini membutuhkan 7 pin digital dimana pin yang digunakan adalah pin D6,D5,D4,D3,D9,D8 dan D7.

5. Penerima Data Digital *Limit Switch*

*Limit Switch* menghasilkan *Output* berupa sinyal digital , dimana data tersebut dapat digunakan sebagai *Input* pada Mikrokontroler ATmega 328. Pada alat ini pin *Output* dari *limit switch* terhubung pada pin D17 .

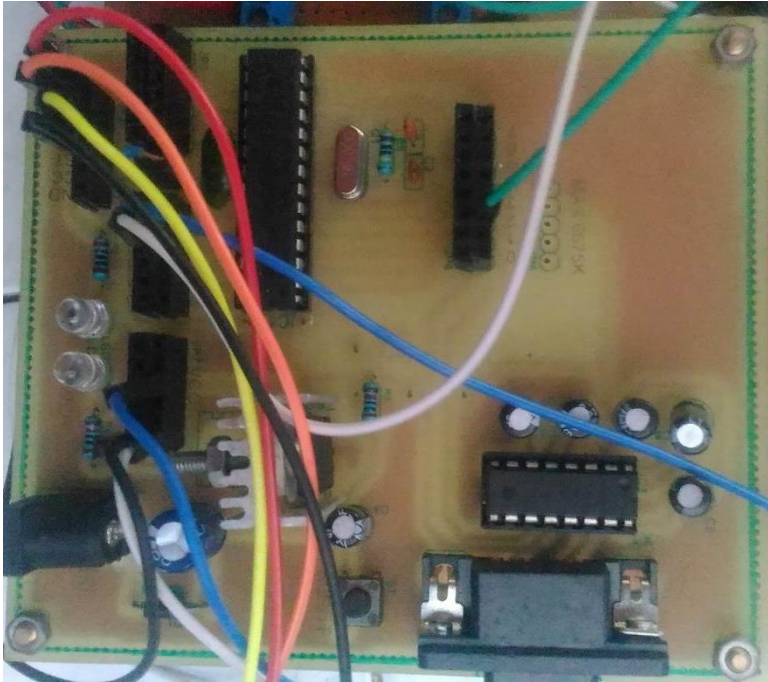
6 Penyimpanan Data Pada *MicroSD*

*MicroSD* digunakan sebagai alat penyimpanan, dimana *microSD* membutuhkan pin SCK,MISO,MOSI dan SS yang terletak pada pin D13,D12,D11,dan D10.

7. Pengirim Data Ke LCD (*Liquid Crystal Display*) dan Penerima Data dari modul RTC (*Real Time Clock*)

LCD dan modul RTC membutuhkan pada alat ini terhubung pada pin SDA (*Serial Data*) dan SCL (*Serial Clock*)

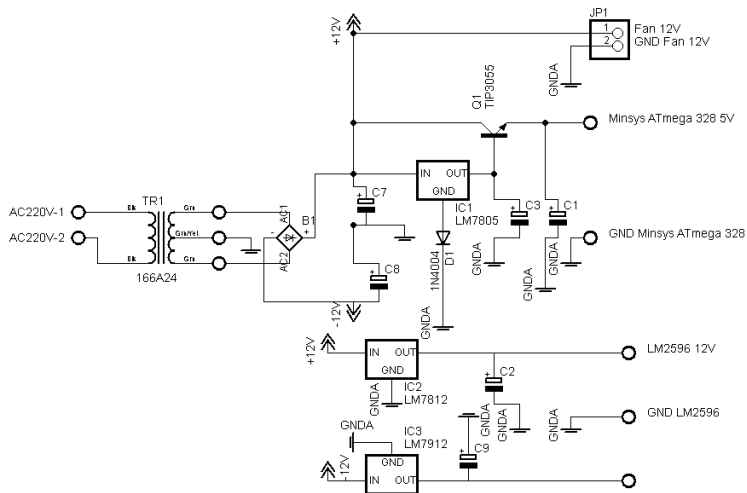
Dari perancangan Mikrokontroler ATmega 328 yang ada pada gambar 3.2 telah dibuat Mikrokontroler ATmega 328 yang nantinya digunakan sebagai pengontrol pada alat yang kami buat.Dimana Mikrokontroler tersebut sudah disesuaikan dengan kebutuhan pada alat yang dibuat.Mikrokontroler yang telah dibuat dapat dilihat pada Gambar 3.3.



**Gambar 3.3** Mikrokontroler ATmega 328

### **3.1.2 Perancangan Rangkaian DC Power Supply**

*DC Power Supply* berfungsi untuk memberikan sumber tegangan pada sistem elektronika . Dalam suatu perangkat elektronika rangkaian *DC Power Supply* ada yang menjadi satu kesatuan dengan perangkat elektronika dan ada juga yang dirancang secara terpisah . Pada Tugas Akhir ini menggunakan rangkaian *DC Power Supply* yang terpisah dengan perangkat elektronika, dengan tegangan keluaran yang dihasilkan terdiri dari 3 besaran , yaitu 5V, 12V dan -12V , rangkaian *DC Power Supply* yang digunakan seperti pada Gambar 3.4.



**Gambar 3.4** Rangkaian DC Power Supply

Pada alat Tugas Akhir Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran tegangan yang dibutuhkan adalah sebesar 5V dan 12V , yaitu tegangan 5V digunakan untuk sumber tegangan Mikrokontroler ATmega 328 dan *Driver Relay*, sedangkan tegangan 12V digunakan sebagai sumber tegangan *solenoid door lock* dan *Coin acceptor*.

Rangkaian DC Power Supply 5V dan 12V ini menggunakan trafo 2 Ampere dan kapasitor elco.. Untuk membatasi tegangan sebesar  $\pm 12V$  maka pada rangkaian DC Power Supply digunakan IC 7812 yang merupakan IC *regulator* tegangan, agar tegangan yang keluar dibatasi sebesar  $\pm 12V$  ketika tegangan masukannya melebihi 12V. Setelah itu terdapat IC 7805 yang juga merupakan IC *regulator* tegangan, yang berfungsi membatasi tegangan sebesar  $\pm 5V$  ketika mendapat tegangan keluaran dari IC 7812 yang besarnya  $\pm 12V$  . Selain itu, fungsi dari IC-regulator 7812 dan 7805 adalah menstabilkan tegangan agar *ouput* tegangan yang keluar tidak jauh dari 12 V dan 5V .

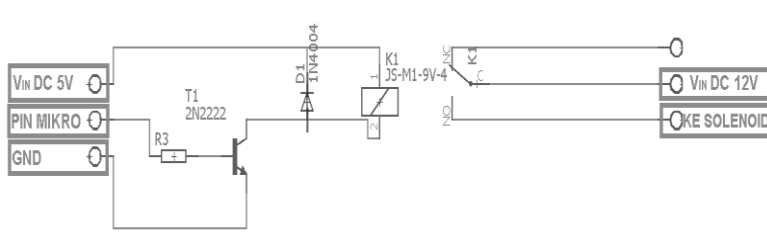
Dari perancangan Rangkaian DC Power Supply diatas, telah dibuat DC Power Supply seperti Gambar 3.5 berikut.



**Gambar 3.5** DC Power Supply

### 3.1.3 Perancangan Driver Relay

Rangkaian *Driver Relay*, seperti Gambar 3.6 merupakan rangkaian untuk mengendalikan atau mengoperasikan perangkat elektronik . Pada alat ini *Driver Relay* berfungsi sebagai pengendali *solenoid door lock* yang dikontrol melalui Mikrokontroler ATmega 328.



**Gambar 3.6** Rangkaian *Driver Relay*

Pada *Driver Relay* diatas terdapat 6 buah pin yang terhubung dengan Mikrokontroler , *Power Supply* dan *solenoid door lock* dengan ketentuan sebagai berikut:

1. Pin X2-1 merupakan VCC dengan tegangan +5V DC dari *Power Supply*
2. Pin X2-2 merupakan pin pemicu yang dihubungkan pada Mikrokontroler
3. Pin X2-3 merupakan GND
4. Pin X1-1 merupakan pin NO (*Normally Open*)
5. Pin X1-2 merupakan pin COM untuk tegangan +12V DC dari *Power Supply*
6. Pin X1-3 merupakan pin NC (*Normally Close*)



Komponen yang digunakan antara lain adalah dioda IN4004 , transistor BC 547 berjenis NPN, dan *relay* 5V .*Relay* yang digunakan pada rangkaian *Driver Relay* ini adalah *relay* dengan jenis HRS4H-S-DC 5V, *relay* jenis ini membutuhkan tegangan +5V DC agar dapat bekerja , dimana tegangan +5V DC didapat dari rangkaian DC *Power Supply*. Pin COM *relay* dihubungkan pada DC *Power Supply* dengan tegangan 12V, dan *solenoid door lock* dihubungkan ke pin NO (*Normally Open*) *relay*.

Cara kerja *Driver Relay* ini adalah apabila pin pemicu *Driver Relay* ini mendapat *Input* berlogika 1 (HIGH) dari *Output* Mikrokontroler ATmega 328 maka akan memicu *basis* transistor BC 547 sehingga arus akan mengalir dari VCC melalui *collector* menuju ke GND melalui *emitter* dan mengaktifkan *coil relay* yang mengakibatkan kondisi *contact* akan berubah dari kondisi NO (*Normally Open*) ke kondisi NC (*Normally Close*), hal tersebut yang menyebabkan tuas *solenoid door lock* memendek (aktif) sehingga pintu loker dapat terbuka. Sedangkan dioda pada rangkaian *Driver Relay* ini berfungsi sebagai pengaman agar tidak terjadi arus balik ke Mikrokontroler ATmega 328.

Dari perancangan Rangkaian *Driver Relay* diatas ,telah dibuat *Driver Relay* seperti pada Gambar 3.7



**Gambar 3.7** *Driver Relay*

### **3.1.4 Setting Coin Acceptor CH-926**

Pada alat ini dilengkapi dengan sistem pembayaran dengan menggunakan *Coin* . Modul yang digunakan untuk mendeteksi *Coin* tersebut adalah modul *Coin acceptor* , *Coin acceptor* bekerja mengenali jenis *Coin* berdasarkan pada bahan, berat dan diameter dari *Coin* tersebut. Pada alat Tugas Akhir ini pembayaran dapat dilakukan dengan menggunakan *Coin* yang berupa uang logam Rp.500 perak dan Rp.500 kuning.. Pin yang digunakan modul *Coin acceptor* CH-926 adalah pin VCC , pin GND dan pin *Coin*. Modul *Coin acceptor* menggunakan

tegangan sebesar +12 DC yang terhubung ke pin VCC dan GND yang didapat dari rangkaian DC *Power Supply*. Sedangkan pin *Coin* digunakan sebagai *Input* pulsa ke Mikrokontroler ATmega 328 Untuk menggunakan *Coin acceptor* maka terlebih dahulu dilakukan setting untuk menentukan jenis *Coin* yang dapat digunakan pada alat ini .

Cara *setting* dari *Coin Acceptor* CH-926 sebagai berikut:

1. Hubungkan Multi *Coin* Sensor CH-926 dengan Vcc sebesar 12V DC, akan terdengar nada ‘beep’ dan LED menyala.
2. Tekan tombol “Add” and “Minus” secara bersamaan selama kurang lebih 3 detik, akan tertampil huruf “A” pada display seven segmen.
3. Tekan tombol “Setup” satu kali, tombol “E” akan tertampil pada display, selanjutnya tentukan jumlah jenis *Coin* dengan menekan tombol “Add” atau “Minus”, kemudian tekan kembali tombol “Setup”.
4. Pada display akan tertampil huruf “H1”, gunakan tombol “Add” dan “Minus” untuk mengatur berapa kali sampling akan dilakukan dengan menggunakan jenis *Coin* pertama, jumlah sampling sebaiknya berkisar 15-20 sampling. Selanjutnya tekan kembali tombol “Setup”.
5. Akan tertampil huruf “P1” pada display, gunakan tombol “Add” dan “Minus” untuk memilih jumlah pulsa yang ingin dikeluarkan pada pin Sinyal keluaran, jika *Coin* jenis pertama terdeteksi. Jumlah pulsa sebaiknya tidak sama untuk tiap jenis *Coin*. Selanjutnya tekan tombol “Setup”.
6. Huruf “F1” akan muncul pada display, gunakan tombol “Add” dan “Minus” untuk mengubah akurasi, nilai akurasi berkisar antara 1-30. Nilai 1 adalah untuk pendeteksian paling akurat, dalam prakteknya nilai 5-10 sudah cukup akurat untuk digunakan. Selanjutnya tekan kembali tombol “Setup”.
7. Sudah selesai untuk *setting* pada chanel pertama, jika jumlah *setting Coin* lebih dari satu maka perlu diulangi langkah ke 4-6 sampai sesuai dengan jumlah *setting Coin*.
8. Tekan kembali tombol “Setup” maka akan tertampil huruf “E” pada display, selanjutnya matikan catu daya. Saat ini *setting* yang dilakukan tadi sudah tersimpan.

Selain dilakukan *setting* juga dilakukan *sampling* untuk mengerjakan cara nomor 4 pada *setting*. Cara melakukan *sampling* adalah sebagai berikut:

1. Menyalakan catu daya.
2. Tekan tombol “Setup” maka akan tertampil huruf “A1” pada display.
3. Mulailah melakukan *sampling* untuk jenis *Coin* pertama, berapa banyak *sampling* tergantung pada *setting* “H1” pada saat melakukan *setting*. Setelah jumlah *sampling* terlampaui maka akan kembali tertampil huruf “A1”.
4. Tekan kembali tombol “Setup”, huruf “A2” akan tertampil. Lakukan *sampling*, kali ini untuk jenis *Coin* kedua.
5. Lakukan kembali langkah ke 4 sampai semua jenis *Coin* sudah disampling, selanjutnya tekan kembali tombol “Setup” maka akan terdengar bunyi ‘beep’ dan *Coin Acceptor* CH-926 sudah siap digunakan.

Pada *setting* yang telah dilakukan jenis *Coin* yang digunakan adalah sebanyak 2 jenis , sehingga pada saat *display seven segment* menampilkan huruf “ E” maka *setting* dilakukan dengan cara menekan tombol “Add” atau “Minus” sampai bernilai 2. Untuk jumlah proses *sampling* yang dilakukan pada proses *sampling* setiap jenis *Coin* adalah sebanyak 15 kali , sehingga pada saat *display seven segment* menampilkan “H1” dan “ H2” dilakukan *setting* dengan menekan tombol “Add” dan “Minus” sampai menunjukkan angka 15. Selanjutnya untuk melakukan *setting* banyaknya pulsa yang dihasilkan oleh *Coin acceptor* CH-926 apabila *Coin* terdeteksi , pada saat *display seven segment* menampilkan “P1” maka dilakukan *setting* terhadap pulsa keluaran *Coin acceptor* dengan menggunakan tombol “Add” dan “Minus”, pada alat ini pulsa yang dipilih adalah 1 untuk kedua jenis *Coin*, sehingga apabila *Coin* yang terdeteksi sesuai maka , Mikrokontroler ATmega 328 akan mendapat pulsa 1

### **3.1.5 Perancangan Connector IC-MAX323 dengan DB9**

Rangkaian IC-MAX232 digunakan sebagai *converter* tegangan untuk menghubungkan antara Mikrokontroler ATmega 328 dengan komputer seperti terlihat pada Gambar 3.8 . Level sinyal Mikrokontroler ATmega 328 adalah TTL (5V) sedangkan pada komputer sinyal

levelnya sebesar 25V atau disebut dengan RS232 oleh karena itu diperlukan *converter* tegangan agar Mikrokontroler ATmega 328 dapat terhubung dengan komputer.

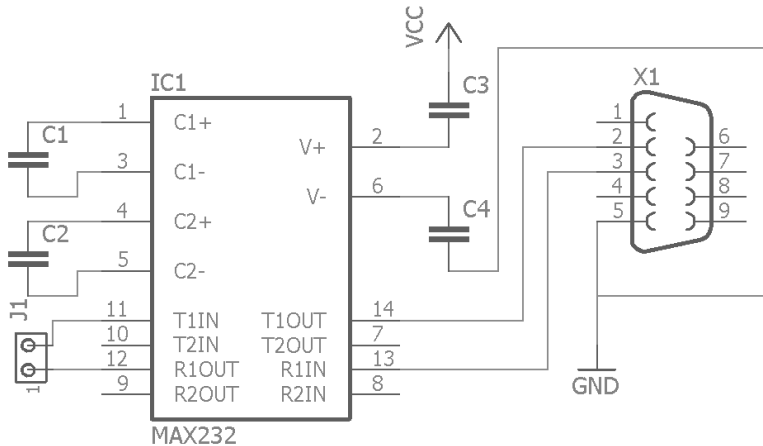
Pada rangkaian yang digunakan terdapat 2 pin yang terhubung dengan Mikrokontroler ATmega 328 antara lain sebagai berikut:

1.Pin Tx

Pin Tx berfungsi sebagai *transmitter* atau pengirim data dari Mikrokontroler ATmega 328 ke IC MAX232

2.Pin Rx

Pin Rx berfungsi sebagai *receiver* atau penerima data IC MAX328 ke Mikrokontroler ATmega 328



**Gambar 3.8** Rangkaian *Connector* IC-MAX232 dengan DB9

Dari perancangan Rangkaian *Connector* IC-MAX323 dengan DB9, telah dibuat *Connector* IC-MAX232 dengan DB9 seperti pada Gambar 3.9



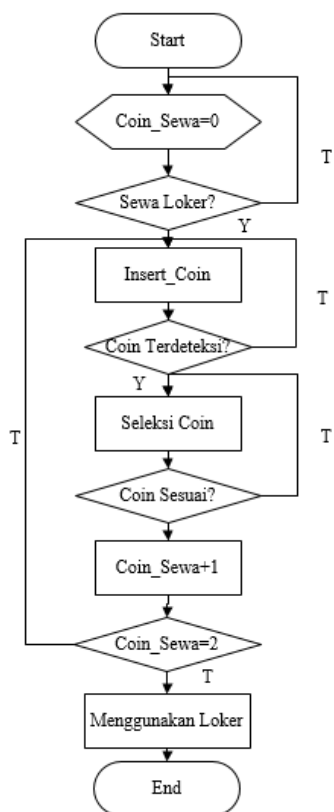
**Gambar 3.9** *Connector IC-MAX232 dengan DB9*

### **3.2 Perancangan Software**

Software yang digunakan untuk alat Tugas Akhir ini adalah pemrograman arduino. Beberapa tahapan pemrograman yang dilakuakn untuk menjalankan alat sistem kunci otomatis pada loker adalah sebagai berikut :

#### **3.2.1 Program Pembayaran**

Pada alat ini terdapat 2 jenis pembayaran , yaitu pembayaran sewa loker dan pembayaran tarif penggunaan. Pembayaran sewa loker adalah sebesar Rp.1000 , pada alat ini *Coin* yang digunakan adalah Rp.500 perak dan Rp.500 kuning , sehingga dibutuhkan 2 *Coin* untuk pembayaran sewa loker. Flowchat dan Program pembayaran sewa loker dapat dilihat pada Gambar 3.10 dan Gambar 3.11 berikut:



**Gambar 3.10** Flowchart Pembayaran Sewa Loker

```

void setup()
{
    pinMode(ledPin, OUTPUT);
    pinMode(interruptPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(interruptPin), insert_coin, RISING);
    Wire.begin();
    lcd.begin(20, 4);
}

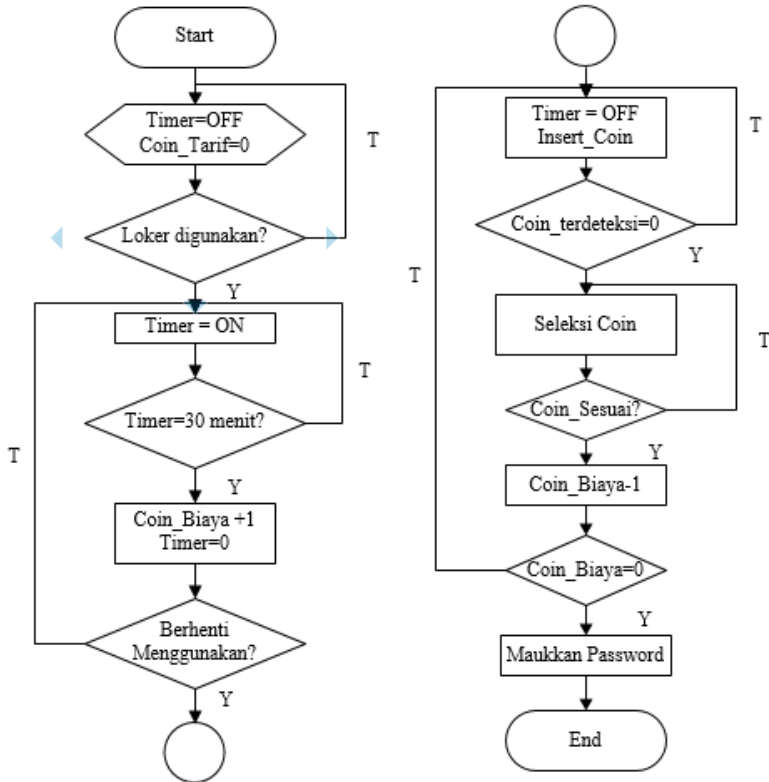
void loop() {
    if(locker==1)
    {
        if(coin_sewa>0)
        {
            lcd.clear();
            lcd.setCursor(2,0);
            lcd.print("MASUKKAN COIN =");
            lcd.setCursor(7,1);
            lcd.print(coin_sewa);
            lcd.print(" COIN");
        }
        else
        {
            locker==2;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("MASUKKAN BARANG =");
        }
    }
}

void insert_coin() {
    if(locker==1)
    {
        coin_sewa=coin_sewa-1;
    }
}

```

**Gambar 3.11** Program Pembayaran Sewa Loker

Sedangkan pembayaran tarif penggunaan berdasarkan lama penggunaan loker , dimana pada alat ini tarif bertambah sebesar RP.500 perak setiap 30 menit , oleh karena itu setiap 30 menit kebutuhan *Coin* bertambah 1 *Coin*. Flowchat dan Program pembayaran tarif penggunaan dapat dilihat pada Gambar 3.12 dan Gambar 3.13 sebagai berikut:



**Gambar 3.12** Flowchart Pembayaran Tarif Penggunaan



---

```

void setup()
{
    pinMode(ledPin, OUTPUT);
    pinMode(interruptPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(interruptPin), insert_coin, RISING);
    Wire.begin();
    lcd.begin(20, 4);}

void loop() {
    if(locker==3)
    {
        if(coin_biaya>0)
        {
            lcd.clear();
            lcd.setCursor(2,0);
            lcd.print("MASUKKAN COIN =");
            lcd.setCursor(7,1);
            lcd.print(coin_biaya);
            lcd.print(" COIN");
        }
        else
        {
            locker==4;
            lcd.clear();
            lcd.setCursor(0,0);
            lcd.print("MASUKKAN PASSWORD =");
        }
    }

    void insert_coin() {
        if (locker==3)
        {
            coin_biaya=coin_biaya-1;
        }
    }
}

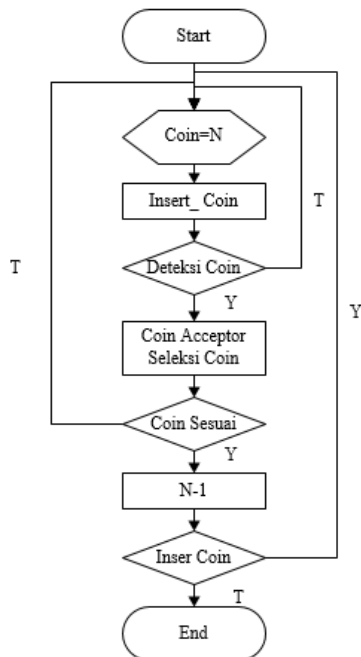
```

---

**Gambar 3.13** Program Pembayaran Tarif Penggunaan

### 3.2.2 Program *Input Coin Acceptor* ke Mikrokontroler ATmega 328

Alat ini menggunakan modul *Coin acceptor* CH-926, dimana modul tersebut memiliki kemampuan untuk mendeteksi jenis-jenis logam, untuk modul *Coin acceptor* CH-926 sendiri memiliki kemampuan mendeteksi 6 jenis *Coin* yang berbeda. Pada alat ini *Coin* yang digunakan hanya dua jenis yaitu uang logam Rp.500 perak dan , uang logam Rp.500 kuning. Modul *Coin acceptor* ini akan memberi pulsa ke Mikrokontroler ATmega 328 sesuai dengan pulsa yang telah ditetapkan pada saat proses setting , dimana kedua jenis *Coin* yaitu uang logam Rp.500 perak dan uang logam Rp.500 kuning telah didedikasikan setting agar memberi pulsa sebanyak 1x pada saat terdeteksi. Sehingga pada saat salah satu jenis *Coin* tersebut terdeteksi , maka akan memberi 1x pulsa pada pin INT0 yang menyebabkan program pada INT0 dijalankan sebanyak 1x setiap kedua jenis *Coin* tersebut terdeteksi. Flowchart dan program *Input sensor Coin acceptor* dapat dilihat pada Gambar 3.14 dan 3.15 sebagai berikut:



**Gambar 3.14** Flowchart *Input Sensor Coin Acceptor*

```

void setup()
{
    attachInterrupt(digitalPinToInterrupt(interruptPin),
    insert_coin, RISING);
}
void loop() {

}

void insert_coin() {
    if(locker==1)
    {
        coin=coin-1;
    }
}

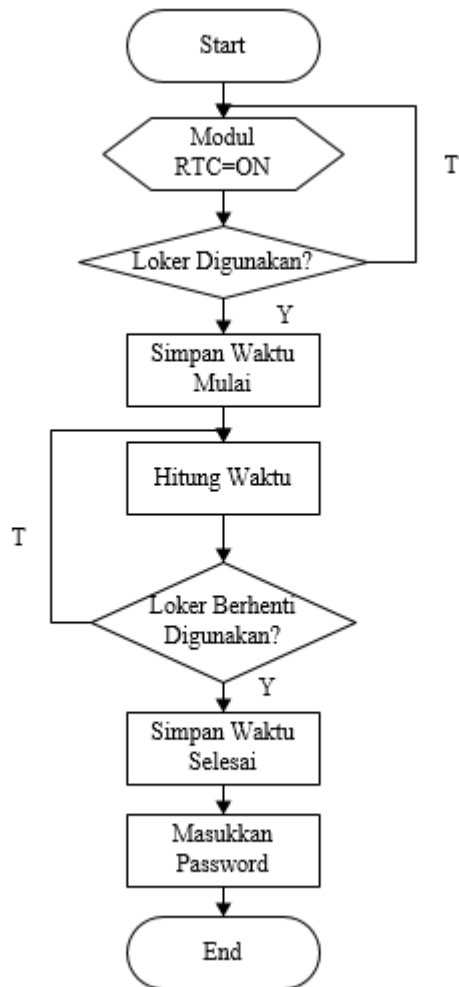
```

**Gambar 3.15** Program *Input Sensor Coin Acceptor*

### 3.2.3 Program Akses Waktu Modul RTC (*Real Time Clock*)

Modul RTC (*Real Time Clock*) merupakan sebuah modul yang dapat menyimpan data jam dan tanggal secara real time , untuk pemakaian pertama kali modul ini harus dilakukan setting terlebih dahulu , agar waktu yang ada didalamnya sesuai dengan eaktu sesungguhnya. Setting waktu tersebut hanya dilakukan satu kali yaitu ketika pemakaian pertama kali,,Selanjutnya waktu pada modul tersebut akan bergulir terus-menerus, selama modul tersebut mendapatkan tegangan.Modul tersebut dilengkapi sebuah baterai yang digunakan sebagai daya masukan.

Pada alat ini RTC digunakan sebagai penyedia waktu yang *Realtime* yang dapat digunakan untuk mengetahui waktu penggunaan loker, dimana waktu yang ada pada modul RTC dapat ditampilkan pada computer melalui komunikasi serial , ataupun disimpan pada *MicroSD*. Pada alat ini waktu yang ada pada loker akan disimpan secara otomatis pada saat loker mulai digunakan dan juga pada saat loker selesai dgunakan. Flowchat dan Program akses waktu modul RTC dapat dilihat pada Gambar 3.16 dan Gambar 3.17 berikut:



**Gambar 3.16** Flowchart Akses Waktu Modul *Real Time Clock*

```

void setup(){
    Wire.begin();
    Serial.begin(9600);
    lcd.begin(20, 4);
    lcd.clear();}
void loop(){
    if(locker==1)
        {Serial.print("Mulai Menggunakan\r\n");
    printDate();
        delay(1000);
        locker=2;}
    else if (locker==2)
        {Serial.print("Selesai Menggunakan\r\n");
    printDate();
        delay(1000);
        locker=3;}
    }
void printDate(){
    Serial.print(monthDay);
    Serial.print("/");
    Serial.print(month);
    Serial.print("/");
    Serial.print(year);
    Serial.print(" ");
    Serial.print(hour);
    Serial.print(":");
    Serial.print(minute);
    Serial.print(":");
    Serial.println(second); }

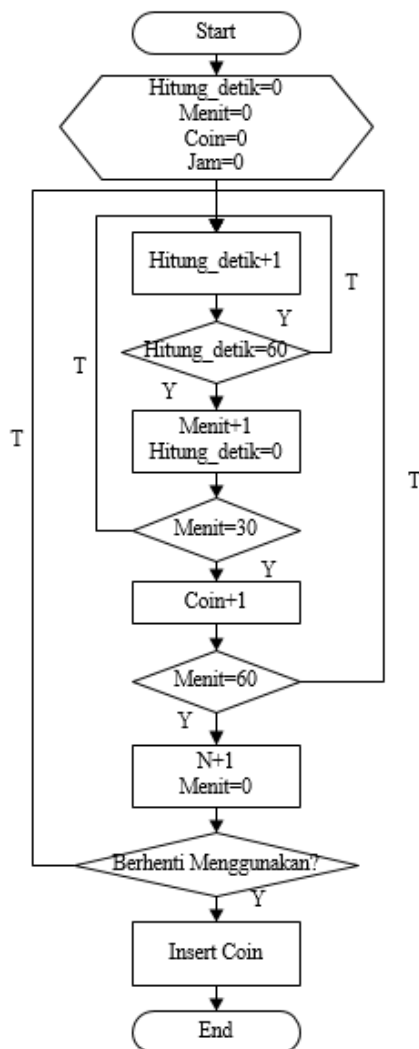
```

**Gambar 3.17** Program Akses Waktu Modul *Real Time Clock*

### 3.2.4 Program *Timer Internal*

*Timer Internal* merupakan *Timer* yang tersedia pada Mikrokontroler ATmega 328 yang dapat diakses dengan cara memberikan program pada Mikrokontroler ATmega 328 .Pada alat Loker Penitipan Otomatis Dilengkapi Dengan Sistem Pembayaran , fungsi dari *Timer Internal* adalah untuk mengetahui lama penggunaan loker dan berapa *Coin* yang dibutuhkan sebagai biaya penggunaan

loker. Flowchart dan program dari *Timer Internal* dapat dilihat pada Gambar 3.18 dan 3.19 sebagai berikut:



**Gambar 3.18** Flowchart *Timer Internal*

```

void loop(){
    if(locker==1)
    {lcd.setCursor(2,0);
    lcd.print("KEBUTUHAN COIN = ");
    lcd.setCursor(7,1);
    lcd.print(coin_biaya);
    lcd.print(" COIN");
    lcd.setCursor (1,2);
    lcd.print("BERHENTI MENYEWA?");
    lcd.setCursor(1,3);
    lcd.print("TEKAN TOMBOL ( * )" ) ;
    }}

void detik(){
    if(locker == 1)
    {hitung_detik++;
    if(hitung_detik ==60)
    {menit++;hitung_detik=0;
    if(menit == 30)
    {coin_biaya=coin_biaya+1;menit=0;
    }
    }
    }
}

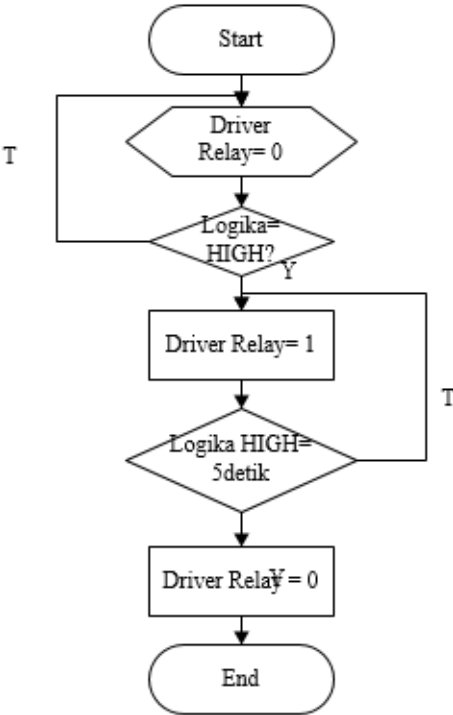
```

**Gambar 3.19** Program *Timer Internal*

### 3.2.5 Program Pengontrol *Driver Relay* untuk *Solenoid Door Lock*

*Driver Relay* pada alat ini digunakan untuk mengendalikan *solenoid door lock* , *Driver Relay* yang digunakan memiliki transistor jenis NPN yang akan aktif apabila pin *basis* pada transistor mendapatkan logika HIGH dari Mikrokontroler ATmega 328, pada saat *basis* mendapatkan logika HIGH arus akan mengalir dari VCC ke Ground sehingga kondisi *coil* terpicu yang menyebabkan *contact* yang awalnya berada pada posisi NO (*Normally Open*) akan berubah ke posisi NC (*Normally Close*). Perubahan tersebut mengakibatkan daya yang terpasang pada pin COM *Relay* akan terhubung ke *contact* NC , dimana pada *contact* NC merupakan *contact* yang digunakan untuk

mengaktifkan *solenoid door lock* , oleh karena itu *solenoid door lock* akan berfungsi (tuas pada *solenoid door lock* yang pada awalnya memanjang akan memendek) ketika *basis* transistor mendapatkan logika HIGH dari Mikrokontroler ATmega 328 dan memperoleh tegangan 12V yang terpasang pada pin COM *Relay*.Namun pengaktifan *solenoid door lock* tidak boleh terlalu lama , karena dapat menyebabkan *solenoid door lock* rusak, oleh karena itu pada program pengendalinya terdapat delay selama 5 detik pada saat logika HIGH yang artinya *solenoid door lock* akan aktif ( tuas memendek) hanya dalam waktu 10 detik sperti pada Gambar flowchart 3.20, sedangkan untuk program pengontrol *Driver Relay solenoid door lock* Dapat dilihat pada Gambar 3.21.



**Gambar 3.20** Flowchart Pengontrol *Driver Relay Solenoid DoorLock*



```

void setup() {
  pinMode(solenoid,OUTPUT);

}

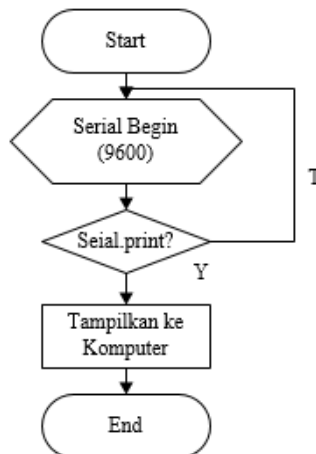
void loop() {
  if (locker==0)
    digitalWrite(solenoid,HIGH);
  delay (5000);
  digitalWrite(solenoid,LOW);
}

```

**Gambar 3.21** Program Pengontrol Driver Relay Solenoid DoorLock

### 3.2.6 Program Komunikasi Serial melalui RS232

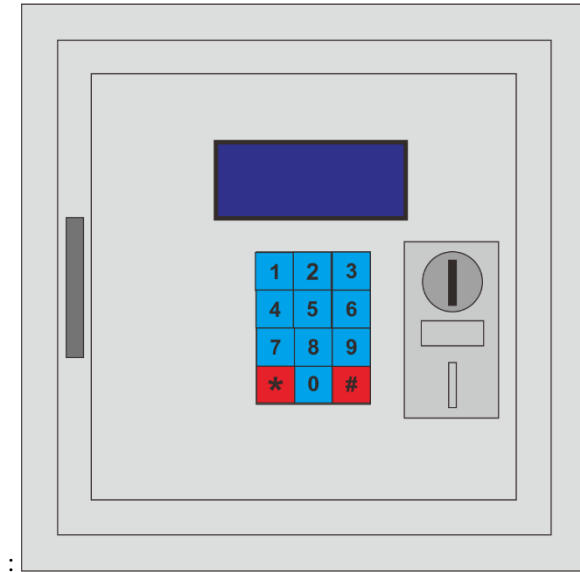
Pada alat ini komunikasi serial digunakan untuk menampilkan data berupa waktu yang berada pada modul RTC ke Komputer , komunikasi serial dilakukan melalui RS232. Untuk menampilkan data pada komputer dilakukan dengan cara menuliskan perintah Serial.print pada program dan data waktu secara otomatis akan ditampilkan.Flowchart menampilkan data ke komputer melalui RS232 dapat dilihat pada Gambar 3.22.



**Gambar 3.22** Flowchart Serial Print Melalui RS232

### 3.3 Desain Loker

Pada Tugas Akhir ini desain alat berupa loker berbentuk persegi dengan ukuran 36cm x 36cm x 36cm dengan ukuran daun pintu 32cm x 32cm. Ketebalan antara daun pintu dengan sisi terluar loker adalah 2cm. Pada daun pintu terdapat gagang pintu yang panjangnya 8cm. Pada bagian daun pintu loker bagian depan akan dipasang LCD 20x4 sebagai tampilan dari kerja loker, *keypad* untuk alat *Input password* dan *Coin acceptor* CH-926 sebagai tempat untuk pembayara. Sedangkan pada bagian belakang loker aka nada lubang sebagai saluran kabel *supply* sebagai penyuplai daya ke rangkaian elektronika, *connector* DB9 to USB untuk menghubungkan Mikrokontroler ke Komputer dan tempat untuk memsasang *MicroSD* sebagai penyimpanan data peminjaman pengguna. Untuk rancangan bentuk loker dapat dilihat pada Gambar-Gambar berikut:



**Gambar 3.23** Bentuk Loker Tampak Depan



**Gambar 3.24** Bentuk Loker Tampak Belakang

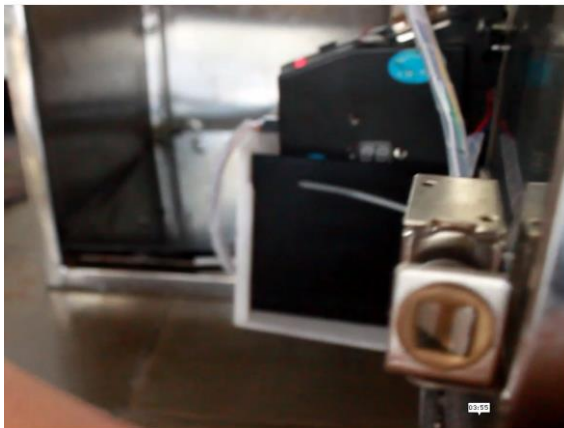


**Gambar 3.25** Bentuk Loker Tampak Samping

Dari Rancangan tersebut telah dibuat sebuah loker seperti pada Gambar-Gambar berikut:



**Gambar 3.26** Loker Tampak Depan



**Gambar 3.27** Pemasangan Solenoid Doorlock



**Gambar 3.28** Penempatan *Coin Acceptor* dan Rangkaian elektronika



**Gambar 3.29** Loker Tampak Samping



**Gambar 3.30** Loker Tampak Belakang

## BAB IV

### HASIL PENGUKURAN DAN PENGUJIAN ALAT

Pada bab ini akan berisi tentang pengujian dan pengukuran terhadap komponen-komponen dan rangkaian elektronika yang digunakan pada Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran. pengujian dan pengukuran dilakukan terhadap tegangan *Output* dari *DC Power Supply*, tegangan *Output* dari Mikrokontroler ATmega 328 , *Timer internal* ATmega 328, data waktu pada modul RTC(*Real Time Clock*) ,tegangan *Input* yang dibutuhkan *Solenoid Doorlock*,serta pengujian terhadap jenis-jenis *Coin* pada *Coin acceptor* CH-926.

#### 4.1 Pengukuran Tegangan *Output DC Power Supply*

Pengukuran tegangan *Output DC Power Supply* dilakukan menggunakan *multimeter* dengan cara menancapkan *probe* positif *multimeter* ke *VCC Power Supply* dan *probe* negatif *multimeter* ke *GND Power Supply*. Pada pengukuran yang dilakukan menggunakan *multimeter* digital dengan data hasil pengukuran pada saat *Power Supply* tanpa beban dan *Power Supply* dengan seluruh komponendan rangkaian elektronika terpasang seperti terlihat pada Tabel berikut:

**Tabel 4.1** Hasil Pengukuran Tegangan *Output DC Power Supply* Tanpa Beban

Tegangan pada <i>output</i> IC regulator 7812	Tegangan pada <i>output</i> IC regulator 7805
11.89 V	5.23 V
11.89 V	5.24 V

**Tabel 4.2** Hasil Pengukuran Tegangan *Output DC Power Supply* Dengan Beban

Tegangan pada <i>output</i> IC regulator 7812	Tegangan pada <i>output</i> IC regulator 7805
11.87 V	4.98 V
11.86 V	4.99 V

Dari kedua Tabel diatas dapat dilihat bahwa tegangan *Output DC Power Supply* dapat dilihat bahwa tegangan *Output* pada IC regulator 7812 pada saat dengan dan tanpa beban memiliki tegangan sebesar  $\pm$  11.86-11.89 V, sehingga tegangan *Output* dari IC regulator 7812 dapat

digunakan sebagai sumber tegangan oleh *Coin Acceptor* yang memerlukan tegangan sebesar 12 V dan *Solenoid Doorlock* yang membutuhkan sumber tegangan sebesar 9-12 V. Sedangkan tegangan *Output* pada IC regulator 7805 pada saat dengan dan tanpa beban adalah sebesar  $\pm 4.98-5.23$ , sehingga tegangan *Output* dari IC regulator 7805 dapat digunakan sebagai sumber tegangan Mikrokontroler ATmega 328 yang tegangan kerjanya sebesar 1.8 – 5.5 V dan *Driver Relay* yang memerlukan tegangan sebesar  $\pm 5$  V

#### 4.2 Pengukuran Tegangan *Output* pin Mikrokontroler ATmega 328

Pengukuran terhadap *Output* pin Mikrokontroler ATmega 328 dilakukan untuk mengetahui besar tegangan *Output* ketika pin Mikrokontroler ATmega 328 diberi logika “HIGH” dan logika “LOW”. pengukuran dilakukan menggunakan *multimeter*, dengan cara menancapkan *probe* positif *multimeter* dengan pin Mikrokontroler ATmega 328 dan *probe* negatif *multimeter* dengan dengan GND Mikrokontroler ATmega 328. Hasil pengukuran tegangan *Output* pin Mikrokontroler ATmega 328 dapat dilihat pada Tabel berikut:

**Tabel 4.3** Hasil Pengukuran Tegangan *Output* pin Mikrokontroler ATmega 328

Logika	Tegangan Output
HIGH	4.91 V
LOW	0 V

Dari Hasil pengujian tegangan *Output* pin Mikrokontroler didapatkan tegangan *Output* pada saat pin Mikrokontroler berlogika “HIGH” sebesar  $\pm 4.91$  V dan sebesar 0 V pada saat pin Mikrokontroler berlogika “LOW”. Oleh karena itu dapat disimpulkan bahwa tegangan *Output* pin Mikrokontroler ATmega 328 yang dibuat bisa digunakan sebagai *Input* modul-modul seperti LCD (*Liquid Crystal Display*) , *Driver Relay*, *Limit Switch*, IC-MAX323 ,modul *MicroSD*,serta modul RTC (*Real Time Clock*) yang membutuhkan tegangan sebesar  $\pm 5$  V.

#### 4.3 Pengukuran Tegangan *Input* yang Diperlukan Pada *Solenoid Doorlock*

Pengukuran terhadap kebutuhan tegangan *Input Solenoid Doorlock* dilakukan untuk mengetahui besar tegangan yang dibutuhkan agar



*Solenoid Doorlock* dapat bekerja. Pengukuran dilakukan dengan menggunakan *multimeter* dan *Power Supply*, dengan cara mengatur besar tegangan pada *Power Supply* dan menghubungkan tegangan *Output Power Supply* ke *Solenoid Doorlock*. Pengukuran tegangan *Input* yang diperlukan *Solenoid Doorlock* dapat dilihat pada Tabel berikut:

**Tabel 4.4** Hasil Pengukuran Tegangan *Input* yang diperlukan *Solenoid Doorlock*

Tegangan (V)	Kondisi Solenoid
5	Tidak Aktif
5,5	Tidak Aktif
6	Tidak Aktif
6,5	Tidak Aktif
7	Tidak Aktif
7,1	Tidak Aktif
7,3	Aktif (Didorong)
7,5	Aktif
8	Aktif
8,5	Aktif
9	Aktif
9,5	Aktif
10	Aktif
10,5	Aktif
11	Aktif
11,5	Aktif
12	Aktif

Dari data yang terdapat pada Tabel diatas dapat dilihat bahwa *Solenoid Doorlock* yang digunakan dapat bekerja normal pada tegangan  $\pm 7.5-12 \text{ V}$ . Oleh karena itu dapat disimpulkan bahwa *Solenoid Doorlock* yang digunakan pada alat ini dapat bekerja dengan menggunakan sumber tegangan *DC Power Supply* yang telah dibuat.

#### 4.4 Pengujian *Timer Internal* Mikrokontroler ATmega 328

Pengujian *Timer Internal* Dilakukan mengetahui kesesuaian *Timer* tersebut dengan waktu yang sebenarnya . Pengujian dilakukan dengan mencocokkan *Timer Internal* Mikrokontroler ATmega 328 dengan *Timer pada Stopwatch*. Pada pengujian tersebut didapatkan hasil perbandingan seperti pada Tabel berikut:

**Tabel 4.5** Hasil Pengujian *Timer Internal* Mikrokontroler ATmega 328

Stopwatch			Timer Arduino Pada LCD		Coin Pada LCD
Detik	Menit	Jam	Detik	Menit	
40	0	0	41	0	0
10	11	0	11	11	0
3	23	0	4	23	0
59	29	0	0	0	1
43	33	0	44	3	1
21	46	0	22	16	1
37	51	0	38	21	1
2	0	1	3	0	2
49	17	1	50	17	2
16	26	1	17	26	2

Dari data yang ada pada Tabel tersebut dapat diketahui bahwa *Timer Internal* Mikrokontroler ATmega 328 sudah sesuai jika dibandingkan dengan waktu yang ada pada *Stopwatch* dimana perbedaan waktunya stabil sebesar 1 detik. Selain itu penambahan jumlah *Coin* yang harus dibayarkan sudah sesuai yaitu 1 *Coin* setiap 30 menit penggunaan.

#### **4.5 Pengujian Modul RTC (*Real Time Clock*) DS 1307**

Pengujian Modul RTC (*Real Time Clock*) DS1307 untuk menyesuaikan data waktu modul RTC (*Real Time Clock*) DS1307 dengan waktu sebenarnya. Pengujian dilakukan dengan cara menampilkan data waktu modul RTC (*Real Time Clock*) pada serial monitor Arduino dengan waktu pada komputer dengan OS windows 10. Hasil dari pengujian dapat dilihat pada Tabel berikut:

**Tabel 4.6** Hasil Modul RTC (*Real Time Clock*) DS 1307

Tampilan Pada Serial Monitor	Tampilan Pada Laptop dengan OS Windows 10	Selisih Waktu
4/6/2017 , 21:0:56	4/6/2017 , 9:00:57 PM	1 detik
4/6/2017 , 21:9:21	4/6/2017 , 9:09:22 PM	1 detik
4/6/2017 , 21:17:34	4/6/2017 , 9:17:35 PM	1 detik
4/6/2017 , 21:23:1	4/6/2017 , 9:23:02 PM	1 detik
4/6/2017 , 21:30:14	4/6/2017 , 21:30:15 PM	1 detik
4/6/2017 , 21:38:42	4/6/2017 , 9:38:43 PM	1 detik
4/6/2017 , 21:46:24	4/6/2017 , 9:46:25 PM	1 detik
4/6/2017 , 21:50:3	4/6/2017 , 9:50:4 PM	1 detik
4/6/2017 , 22:1:31	4/6/2017 , 10:01:32 PM	1 detik
4/6/2017 , 22:13:29	4/6/2017 , 10:13:30 PM	1 detik

Dari data hasil pengujian pada Tabel diatas dapat dilihat bahwa waktu yang ada pada modul RTC (*Real Time Clock*) berselisih sebesar 1 detik, dimana selisih tersebut tetap sama dari waktu ke waktu. Oleh karena itu modul RTC (*Real Time Clock*) yang digunakan sudah memiliki waktu yang sesuai dengan waktu sebenarnya.

#### 4.6 Pengujian Sensor *Coin Acceptor* CH-926

Pengujian sensor *Coin Acceptor* CH-926 dilakukan untuk mengetahui bahwa sensor ini dapat mendeteksi *Coin* yang digunakan sesuai dengan yang telah diatur atau tidak, dimana pada alat Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran *Coin* yang digunakan adalah uang logam Rp.500 perak dan Rp.500 kuning. Oleh karena itu dilakukan pengujian terhadap jenis-jenis uang logam yang ada. Pengujian dilakukan dengan cara memasukkan *Coin* yang berupa uang logam ke *Coin Acceptor* CH-926. Apabila jenis *Coin* yang dimasukkan sesuai, maka kebutuhan *Coin* yang ditampilkan pada LCD akan berkurang sebesar 1 *Coin*. Hasil pengujian terhadap *Coin* dapat dilihat pada Tabel berikut:

**Tabel 4.7** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.100 Perak

Percobaan	Tampilan LCD	Keterangan
1	10 coin	Tampilan Awal
2	10 coin	1 Coin Ditambahkan
3	10 coin	1 Coin Ditambahkan
4	10 coin	1 Coin Ditambahkan
5	10 coin	1 Coin Ditambahkan
6	10 coin	1 Coin Ditambahkan
7	10 coin	1 Coin Ditambahkan
8	10 coin	1 Coin Ditambahkan
9	10 coin	1 Coin Ditambahkan
10	10 coin	1 Coin Ditambahkan

**Tabel 4.8** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.100 Kuning

Percobaan	Tampilan LCD	Keterangan
1	10 coin	Tampilan Awal
2	10 coin	1 Coin Ditambahkan
3	10 coin	1 Coin Ditambahkan
4	10 coin	1 Coin Ditambahkan
5	10 coin	1 Coin Ditambahkan
6	10 coin	1 Coin Ditambahkan
7	10 coin	1 Coin Ditambahkan
8	10 coin	1 Coin Ditambahkan
9	10 coin	1 Coin Ditambahkan
10	10 coin	1 Coin Ditambahkan

**Tabel 4.9** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.200 Perak

Percobaan	Tampilan LCD	Keterangan
1	10 coin	Tampilan Awal
2	10 coin	1 Coin Ditambahkan
3	10 coin	1 Coin Ditambahkan
4	10 coin	1 Coin Ditambahkan
5	10 coin	1 Coin Ditambahkan
6	10 coin	1 Coin Ditambahkan
7	10 coin	1 Coin Ditambahkan
8	10 coin	1 Coin Ditambahkan
9	10 coin	1 Coin Ditambahkan
10	10 coin	1 Coin Ditambahkan

**Tabel 4.10** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.500 Perak

Percobaan	Tampilan LCD	Keterangan
1	coin 10	Tampilan Awal
2	coin 9	1 Coin Ditambahkan
3	coin 8	1 Coin Ditambahkan
4	coin 7	1 Coin Ditambahkan
5	coin 6	1 Coin Ditambahkan
6	coin 5	1 Coin Ditambahkan
7	coin 4	1 Coin Ditambahkan
8	coin 3	1 Coin Ditambahkan
9	coin 2	1 Coin Ditambahkan
10	coin 1	1 Coin Ditambahkan

**Tabel 4.11** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.500 Kuning

Percobaan	Tampilan LCD	Keterangan
1	coin 10	Tampilan Awal
2	coin 9	1 Coin Ditambahkan
3	coin 8	1 Coin Ditambahkan
4	coin 7	1 Coin Ditambahkan
5	coin 6	1 Coin Ditambahkan
6	coin 5	1 Coin Ditambahkan
7	coin 4	1 Coin Ditambahkan
8	coin 3	1 Coin Ditambahkan
9	coin 2	1 Coin Ditambahkan
10	coin 1	1 Coin Ditambahkan

**Tabel 4.12** Hasil Pengujian *Coin Acceptor* Dengan *Coin* Rp.1000 Perak

Percobaan	Tampilan LCD	Keterangan
1	coin 10	Tampilan Awal
2	coin 10	1 Coin Ditambahkan
3	coin 10	1 Coin Ditambahkan
4	coin 10	1 Coin Ditambahkan
5	coin 10	1 Coin Ditambahkan
6	coin 10	1 Coin Ditambahkan
7	coin 10	1 Coin Ditambahkan
8	coin 10	1 Coin Ditambahkan
9	coin 10	1 Coin Ditambahkan
10	coin 10	1 Coin Ditambahkan

Dari hasil pengujian terhadap berbagai jenis *Coin* berupa uang logam dapat dilihat pada Tabel-Tabel diatas bahwa hanya uang logam Rp.500 kuning dan Rp.500 perak yang dapat diterima oleh *Coin Acceptor* CH-926 ,dimana perunahan dapat dilihat dari jumlah *Coin* yang semakin berkurang setiap uang logam Rp.500 kuning dan Rp.500 perak di tambahkan pada *Oina Acceptor* CH-926 ini. Sedangkan untuk uang logam lainnya tidak dapat digunakan untuk proses pembayaran pada alat ini.Hal tersebut dikarenakan hanya uang logam Rp.500 kuning dan Rp.500 perak yang sudah di lakukan proses *setting* dan *sampling* pada *Coin Acceptor* CH-926 yang digunakan pada alat ini.

#### 4.7 Pengujian Tampilan Pada *Serial Monitor*

Pengujian ini dilakukan untuk mengetahui tampilan pada *Serial Monitor* apakah sudah sesuai dengan apa yang ada pada program.Pengujian dilakukan dengan cara memberikan perintah serial.print pada program dan mengubah nilai *baud rate* pada *Serial Monitor* maupun program yang digunakan.Hasil pengujian dapat dilihat pada Tabel 4.13 sebagai berikut:

**Tabel 4.13** Hasil Pengujian Tampilan *Serial Monitor*

Tampilan Pada Program	Baud Rate		Tampilan Serial Monitor
	Program	Serial Monitor	
12345 ABCDE	9600	9600	12345 ABCDE
12346 ABCDE	9600	19200	☐x☐☐☐
12347 ABCDE	115200	115200	12345 ABCDE
12348 ABCDE	250000	115200	%á /Ĭ

Dari hasil pengujian yang terdapat pada Tabel diatas dapat disimpulkan bahwa tampilan pada *Serial Monitor* akan sesuai dengan yang ada pada program apabila *baud rate* antara program dan *Serial Monitor* memiliki besaran yang sama.Oleh karena itu untuk mendapatkan tampilan yang sesuai maka dilakukan *setting* terhadap *baud rate* baik itu pada program maupun pada *Serial Monitor*.

#### 4.8 Pengujian Kinerja Alat Keseluruhan

Pada pengujian kinerja alat keseluruhan dilakukan untuk menguji tingkat keberhasilan alat ketika dioperasikan secara berulang-

ulang,dimana pengujian dilakukan terhadap beberapa fitur-fitur yang ada Bagian fitur yang diuji antara lain:

#### 4.8.1 Fitur *Inset Coin*

Pada pengujian terhadap fitur *Inset Coin* dapat dikatakan berhasil apabila *Coin Acceptor* dapat menjalankan program yang ada apabila ditambahkan *Coin* yang sesuai yaitu *Coin* Rp.500 perak dan *Coin* Rp.500 kuning dan menolak ketika *Coin* yang dimasukkan tidak sesuai.Adapun salah satu contoh proses pengujian terhadap fitur *Inset Coin*,dimana pada awalnya pada layar LCD(*Liquid Crystal Display*) ditampilkan jumlah *Coin* yang harus ditambahkan pengguna seperti Gambar 4.1



**Gambar 4.1**Tampilan Kebutuhan Jumlah *Coin*

Kemudian pada saat pengguna menambahkan jenis *Coin* yang sesuai yaitu *Coin* Rp.500 perak dan Rp.500 kuning maka jumlah kebutuhan *Coin* yang ada pada LCD(*Liquid Crystal Display*) akan berkurang seperti pada Gambar 4.2 ,sedangkan apabila *Coin* yang ditambahkan tidak sesuai maka jumlah kebutuhan *Coin* tidak akan berkurang serta *Coin* yang ditambahkan akan keluar kembali dari *Coin Acceptor* seperti pada Gambar 4.3.



**Gambar 4.2** Tampilan Ketika *Coin* yang Ditambahkan Sesuai



**Gambar 4.3** Tampilan Ketika *Coin* yang Ditambahkan Tidak Sesuai

#### 4.8.2 Fitur *Timer* dan Tarif Penggunaan

Pada Pengujian pada fitur *Timer* dan tarif penggunaan dapat dikatakan berhasil apabila banyaknya kebutuhan *Coin* sesuai dengan lama penggunaan alat ini, dimana setiap waktu penggunaan selama 30 menit maka jumlah kebutuhan *Coin* akan bertambah sebanyak 1 *Coin*. Salah satu contoh pengujian terhadap *Timer* dan tarif penggunaan dapat dilihat pada Gambar 4.4.





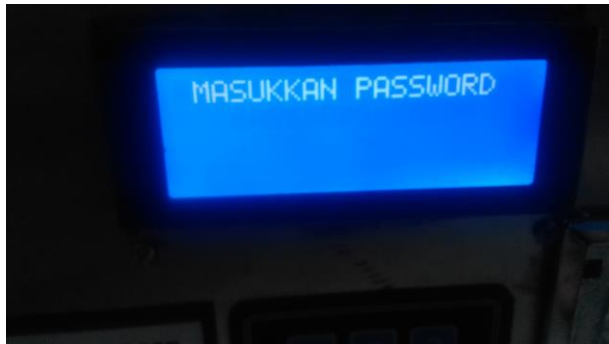
**Gambar 4.4** Tampilan *Timer* dan *Kebutuhan Coin*

Pada Gambar tersebut terlihat waktu penggunaan adalah 1 jam dan kebutuhan *Coin* sebanyak 2 *Coin*. Sehingga pada pengujian tersebut jumlah kebutuhan *Coin* dan lama penggunaan sesuai.

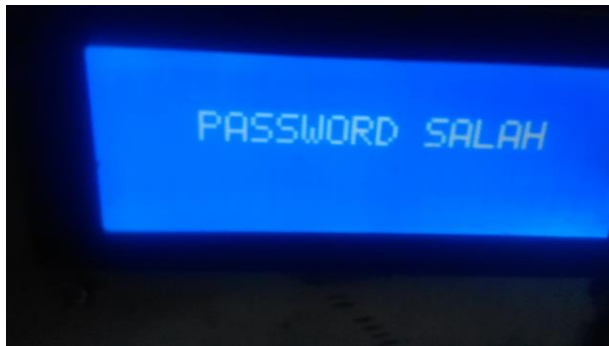
#### **4.8.3 Fitur *Password***

Pada pengujian terhadap fitur *Password* dapat dikatakan berhasil apabila ketika melakukan pencocokan *Password* awal penggunaan dengan *Password* akhir benar. Untuk mengetahui hal tersebut, maka pada proses pengujian pertama dilakukan *Input Password* awal, setelah itu pada tahap akhir dilakukan *Input Password* akhir. Apabila *Password* akhir sesuai dengan *Password* awal maka pada akan tampil “PASSWORD BENAR” pada LCD (*Liquid Crystal Display*), dan apabila tidak sesuai maka tampilan pada LCD (*Liquid Crystal Display*) adalah “PASSWORD SALAH”.

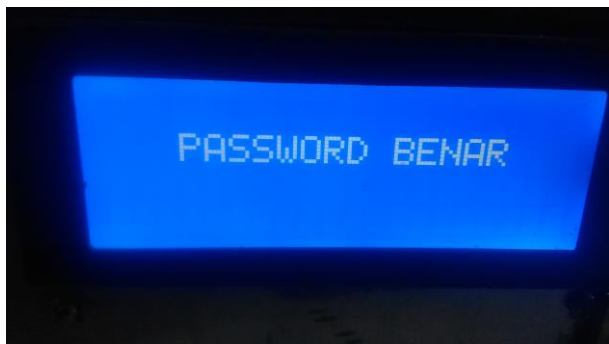
Salah satu pengujian yang dilakukan dapat dilihat pada Gambar-Gambar dibawah, dimana awalnya terdapat perintah untuk memasukkan *Password* seperti pada Gambar 4.5. Setelah itu dimasukkan *Password* yang tidak sesuai maka tampilan “PASSWORD SALAH” akan muncul, seperti gambar 4.6 dan pengguna dipersilahkan untuk memasukkan kembali *Password* yang benar. Ketika *Password* yang dimasukkan sesuai maka tampilan yang muncul adalah “PASSWORD BENAR” seperti pada Gambar 4.7



**Gambar 4.5** Tampilan Perintah Memasukkan *Password*



**Gambar 4.6** Tampilan Ketika *Password* Salah



**Gambar 4.7** Tampilan Ketika *Password* Benar

Untuk mendapatkan persentase keberhasilan terhadap kinerja alat, dilakukan pengujian terhadap ketiga fitur diatas secara berulang-ulang, dimana apabila disetiap fitur terjadi kesalahan maka akan dianggap sebagai *error* dan apabila pada setiap fitur tidak terjadi kesalahan maka dianggap berhasil. Hasil pengujian kinerja alat keseluruhan dapat dilihat pada Tabel 4.14

**Tabel 4.14** Hasil Pengujian Kinerja Alat Keseluruhan

NO	Pangujian Fitur			Keterangan
	<i>Insert Coin</i>	<i>Timer Penggunaan</i>	<i>Password</i>	
1	Berhasil	Berhasil	Berhasil	-
2	Berhasil	Berhasil	Berhasil	-
3	Berhasil	Berhasil	Berhasil	-
4	Berhasil	Berhasil	Berhasil	-
5	Berhasil	Berhasil	Berhasil	-
6	Berhasil	Berhasil	Berhasil	-
7	Berhasil	Berhasil	Berhasil	-
8	<i>Error</i>	Berhasil	Berhasil	Koin Rp.500 Perak Tidak diterima
9	Berhasil	Berhasil	Berhasil	-
10	Berhasil	Berhasil	Berhasil	-
11	<i>Error</i>	Berhasil	Berhasil	Koin Rp.500 Perak Tidak diterima
12	Berhasil	Berhasil	Berhasil	-
13	Berhasil	Berhasil	Berhasil	-
14	Berhasil	Berhasil	Berhasil	-
15	Berhasil	Berhasil	Berhasil	-
16	Berhasil	Berhasil	Berhasil	-
17	Berhasil	Berhasil	Berhasil	-
18	Berhasil	Berhasil	Berhasil	-
19	Berhasil	<i>Error</i>	Berhasil	Kebutuhan <i>Coin</i> Tidak Sesuai Timer
20	Berhasil	Berhasil	Berhasil	-
21	Berhasil	Berhasil	Berhasil	-
22	Berhasil	Berhasil	Berhasil	-
23	<i>Error</i>	Berhasil	Berhasil	Koin Rp.500 Perak Tidak diterima
24	Berhasil	Berhasil	Berhasil	-

25	Berhasil	Berhasil	Berhasil	-
26	Berhasil	Berhasil	Berhasil	-
27	Berhasil	Berhasil	Berhasil	-
28	Berhasil	Berhasil	Berhasil	-
29	Berhasil	Berhasil	Berhasil	-
30	Berhasil	Berhasil	Berhasil	-
31	Berhasil	Berhasil	Berhasil	-
32	Berhasil	Berhasil	Berhasil	-
33	Berhasil	Berhasil	Berhasil	-
34	Berhasil	Berhasil	Berhasil	-
35	Berhasil	Berhasil	Berhasil	-
36	<i>Error</i>	Berhasil	Berhasil	Koin Rp.500 Perak Tidak diterima
37	Berhasil	Berhasil	Berhasil	-
38	Berhasil	Berhasil	Berhasil	-
39	Berhasil	Berhasil	Berhasil	-
40	Berhasil	Berhasil	Berhasil	-

Dari hasil pengujian kinerja alat secara keseluruhan yang dilakukan sebanyak 40 kali dapat dilihat pada tabel 4.13 bahwa terjadi lima kali *error*. *Error* yang terjadi disebabkan jumlah *Coin* yang tidak sesuai dengan lama penggunaan serta *Coin* yang tidak dapat diterima oleh modul *Coin Acceptor* CH-926. Dari hasil pengujian tersebut persentase keberhasilan sebesar  $(35 / 40) \times 100\% = 87.5 \%$ . dengan persentase *error* sebesar  $100\% - 87.5\% = 12.5\%$ .

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari hasil pengukuran dan pengujian terhadap komponen dan rangkaian elektronika yang digunakan pada alat Loker Penitipan Otomatis Dilengkapi Sistem Pembayaran dapat disimpulkan bahwa:

1. Tegangan *Output* yang dihasilkan oleh IC *regulator* 7805 pada saat terpasang beban lebih kecil dari pada Tegangan *Output* pada saat beban tidak terpasang.
2. Tegangan *Output* pada pin Mikrokontroler ATmega 328 yang digunakan pada alat ini sebesar 4.91V pada saat diberi logika “HIGH” dan 0V pada saat diberi logika “LOW”.
3. Tegangan yang diperlukan untuk mengaktifkan *Solenoid Doorlock* yang digunakan pada alat ini adalah 7.3V namun pada tegangan tersebut kinerja *Solenoid Doorlock* untuk menarik tuas masih pelan. Kinerja *Solenoid Doorlock* mulai normal ketika diberi tegangan 7.5V.
4. *Timer Internal* pada Mikrokontroler ATmega 328 yang digunakan pada alat ini sudah sesuai dengan waktu yang ditunjukkan *Stopwatch*. Mikrokontroler ATmega 328 dan *Stopwatch* yang tidak bersamaan
5. Pada pengujian terhadap jenis-jenis uang logam yang ada terhadap sensor *Coin Acceptor* CH-926 . uang logam yang dapat digunakan dan terdeteksi oleh sensor tersebut adalah uang logam Rp.500 perak dan Rp.500 kuning. Hal tersebut dikarenakan kedua jenis uang logam tersebut sudah dilakukan proses *setting* dan *sampling* pada sensor *Coin Acceptor* CH-926
6. Pada pengujian tampilan serial monitor akan sesuai apabila baud rate pada program dan serial monitor bernilai sama.
7. Pada pengujian kerja alat secara keseluruhan didapatkan persentase keberhasilan sebesar 87.5% dengan persentase *error* sebesar 12.5%.

#### **5.2 Saran**

Untuk pengembangan alat sejenis selanjutnya, mungkin dapat dilakukan dengan cara menambahkan beberapa jenis uang logam yang bisa digunakan atau juga dapat menambahkan modul yang bisa

mendeteksi uang kertas sehingga lebih mempermudah pengguna untuk melakukan proses pembayaran.

## DAFTAR PUSTAKA

- [1] \_\_\_\_,2014, “Tempat Umum”,  
[https://id.wikipedia.org/wiki/Tempat umum](https://id.wikipedia.org/wiki/Tempat_umum) diakses Tanggal 9 April 2017.
- [2] Setyadi R T,Badzlina A., “Perancangan Prototype Sistem Kunci Otomatis Pada Loker Penitipan Barang Komersial”, *Tugas Akhir*, Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [3] \_\_\_\_\_. *8-bit AVR Microcontroller with 4/8/16/32K Bytes In-System Programmable*. Atmel Corporation, USA, 2009.
- [4] Jurnal Politeknik Sriwijaya. BAB II Tinjauan Pustaka.  
<http://eprints.polsri.ac.id/2069/3/BAB%20II.pdf>, diakses 9 April 2017
- [5] Rakhmatsyah A,Suwastika N A,Haryanto,R. “Implementasi Sistem Kunci Pintu Otomatis Untuk Smart Home Menggunakan SMS Gateway”,*Jurnal*,Telkom Univercity , 2015
- [6] \_\_\_\_, “Multi Coin Acceptor/Selector/Sensor CH-926”, DIY-Tech. 2012.
- [7] \_\_\_\_,”Manual of CH-926”,DIY-Tech, 2012.
- [8] \_\_\_\_,2012,”LCD (Liquid Cristal Display”, <https://elektronika-dasar.web.id/lcd-liquid-cristal-display/> diakses tanggal 10 April 2017.
- [9] \_\_\_\_,2013,”Rangkaian Driver Relay”,  
<http://www.rangkaianelektronika.org/rangkaian-driver-relay.html>  
diakses tanggal 10 April 2017.
- [10]\_\_\_\_,2015,”Bagaimana Cara Membuat Driver Relay”,  
<https://utakatikmikro.com/2015/03/04/bagaimana-cara-membuat->

driver-relay-untuk-menggerakan-kontaktor-3phase-motor-dll-untuk-mikrokontroler/ diakses Tanggal 29 Juli 2017.

- [11] \_\_\_\_\_.2012.*Limit Switch dan Saklar Push-on*. <http://elektronika-dasar.web.id/limit-switch-dan-saklar-push-on/> (Diakses Tanggal 30 Juni 2017).
- [12] Suyadi,"Komunikasi Serial dan Port Serial (COM)".*Jurnal Teknik Informatika UMS*. Surakarta .2012.
- [13]\_\_\_\_\_,2010.*MicroSD*, <https://id.wikipedia.org/wiki/MicroSD> diakses tanggal 11 April 2017.
- [14] Wijayanto I,"Pemrograman RTC DS1307 menggunakan CV AVR", <http://iwijayanto.staff.telkomuniversity.ac.id/pemrograman-rtc-ds1307-menggunakan-cv-avr/> diakses tanggal 11 April 2017.
- [15]\_\_\_\_\_.2015.*Pemrograman Arduino Berbasis Bahasa C*. <http://www.sebatekno.com/pemrograman-arduino-berbasis-bahasa-c> diakses tanggal 11 April 2017.



## LAMPIRAN

### A. Listing Program Keseluruhan

```
#include <EEPROM.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <TimerOne.h>
#include <SPI.h>
#include <SD.h>
#define DS1307_ADDRESS 0x68
File myFile;
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3,
POSITIVE);
const byte Solenoid = 16;
const byte Limit = 15;
const byte interruptPin = 2;
const byte ROWS = 4;
const byte COLS = 3;
const int chipSelect = 10;
byte zero = 0x00;
int second,minute,hour,weekDay,monthDay,month,year ;
int biaya,sewa;
int Coin_biaya;
int Coin_sewa=0;
int status_locker=0;
int hitung_detik,menit,SetengahJam,Jam;
char key;
char pU1,pU2,pU3,pU4,pU5,pU6;
char pu1,pu2,pu3,pu4,pu5,pu6;
char pA1,pA2,pA3,pA4,pA5,pA6;
char pa1,pa2,pa3,pa4,pa5,pa6;
int cnt,cnt_pass;
int i = 0;
int j = 10;
int k = 30;
int x = 7;
char keys[ROWS][COLS] = {{ '1','2','3'},
```

```

{'4','5','6'},
{'7','8','9'},
{'*','0','#'}
};
byte rowPins[ROWS] = {6,5,4,3};
byte colPins[COLS] = {9,8,7};
Keypad keypad = Keypad( makeKeymap(keys), rowPins,
colPins, ROWS, COLS );
void setup()
{
EEPROM.write(21,'1');
EEPROM.write(22,'1');
EEPROM.write(23,'0');
EEPROM.write(24,'3');
EEPROM.write(25,'9');
EEPROM.write(26,'6');
pinMode(Solenoid, OUTPUT);
pinMode(Limit, INPUT);
pinMode(interruptPin, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(interruptPin),
insert_Coin, RISING);
Wire.begin();
Serial.begin(9600);
lcd.begin(20, 4);
Timer1.initialize(1000000);
Timer1.attachInterrupt(detik);
}
void loop() {
key = keypad.getKey();
Jam=EEPROM.read(9);
menit=EEPROM.read(10);Coin_biaya=EEPROM.read(8);
Coin_sewa=EEPROM.read(7);
status_locker = EEPROM.read(0);
pU1=EEPROM.read(1);
pU2=EEPROM.read(2);
pU3=EEPROM.read(3);
pU4=EEPROM.read(4);
pU5=EEPROM.read(5);
pU6=EEPROM.read(6);

```

```

pu1=EEPROM.read(11);
pu2=EEPROM.read(12);
pu3=EEPROM.read(13);
pu4=EEPROM.read(14);
pu5=EEPROM.read(15);
pu6=EEPROM.read(16);
pA1=EEPROM.read(21);
pA2=EEPROM.read(22);
pA3=EEPROM.read(23);
pA4=EEPROM.read(24);
pA5=EEPROM.read(25);
pA6=EEPROM.read(26);
pa1=EEPROM.read(31);
pa2=EEPROM.read(32);
pa3=EEPROM.read(33);
pa4=EEPROM.read(34);
pa5=EEPROM.read(35);
pa6=EEPROM.read(36);
cnt_pass=EEPROM.read(17);
if(status_locker==255)
{
  lcd.setCursor(3,0);
  lcd.print("LOKER PENITIPAN");
  lcd.setCursor(1,1);lcd.print("OTOMATIS DILENGKAPI");
  lcd.setCursor(2,2);
  lcd.print("SISTEM PEMBAYARAN");
  lcd.setCursor(3,3);
  lcd.print("GUNAKAN ? ( * )");
  key=keypad.getKey();
  if(key != NO_KEY)
  {
    if(key=='*')
    { lcd.clear();
      EEPROM.write(7,0);
      EEPROM.write(0,0);
    }
  }
}
else if(status_locker==0)

```

```

{
if (!SD.begin(10))
{
return;
delay (1000);
EEPROM.write(0,1);
}
else
{
EEPROM.write(0,1);
}
}
else if(status_locker==1)
{
if(Coin_sewa<2)
{
lcd.setCursor(0,0);
lcd.print("MASUKKAN 2 COIN 500");lcd.setCursor(2,1);
lcd.print("UNTUK BIAYA SEWA");
lcd.setCursor(2,2);
lcd.print("COIN DIBUTUHKAN ");
lcd.setCursor(7,3);
lcd.print((Coin_sewa*-1)+2);
lcd.print(" COIN");
EEPROM.write(7,Coin_sewa);
}
else
{
lcd.clear();
lcd.setCursor(2,1);
lcd.print("TEKAN (*) UNTUK");
lcd.setCursor(3,2);
lcd.print("MEMBUKA LOKER");
delay(100);
if(key != NO_KEY)
{
if(key=='*')
{ lcd.clear();
lcd.setCursor(0,1);

```

```

lcd.print("MASUKKAN BARANG ANDA");
digitalWrite(Solenoid,HIGH);
delay(5000);
lcd.clear();
digitalWrite(Solenoid,LOW);
EEPROM.write(0,2);
delay(1000);
}
}
}
}
else if (status_locker==2)
{ if(digitalRead(Limit)==LOW)
{ lcd.clear();
lcd.setCursor(0,1);
lcd.print("TUTUP KEMBALI LOKER");
cnt=0;
delay (100);
lcd.clear();
}
else
{
lcd.setCursor(1,0);
lcd.print("MASUKKAN PASSWORD");
if(cnt<6)
{
if(key != NO_KEY)
{
lcd.setCursor(x,1);
lcd.print(key);
x++;
i++;
cnt++;
EEPROM.write(i,key);
}
}
else
{
lcd.setCursor(1,3);

```

```

lcd.print("SIMPAN PASSWORD (*");
if(key != NO_KEY)
{
if (key=='*')
{
lcd.clear();
lcd.setCursor(1,1);
lcd.print("PASSWORD TERSIMPAN");delay (3000);
lcd.clear();
Serial.print("LOKER MULAI DIGUNAKAN
PADA \r\n");
printDate();
MicroSD();
delay(1000);
Jam=0;
menit=0;
hitung_detik=0;
x=7;
cnt=0;
i=0;
EEPROM.write(8,0);
EEPROM.write(9,0);
EEPROM.write(10,0);
EEPROM.write(0,3);
}
}
}
}
}
else if(status_locker==3)
{
lcd.setCursor(2,0);
lcd.print("KEBUTUHAN COIN = ");
lcd.setCursor(7,1);
lcd.print(Coin_biaya);
lcd.print(" COIN");
lcd.setCursor (0,2);
lcd.print(Jam);
lcd.print(" jam ") ;

```

```

lcd.print(menit);
lcd.print(" mnt ");lcd.print(hitung_detik);
lcd.print(" dtk");
lcd.setCursor(3,3);
lcd.print("BERHENTI( * )");
delay(100);
if(key != NO_KEY)
{
if(key=='*')
{
delay(1000);
lcd.clear();
EEPROM.write(0,4);
}
}
else if(status_locker==4)
{
if(Coin_biaya>0)
{
lcd.setCursor(2,0);
lcd.print("MASUKKAN COIN 500");
lcd.setCursor(2,1);
lcd.print("UNTUK PENGGUNAAN ");
lcd.setCursor(2,2);
lcd.print("SEBANYAK ");
lcd.print(Coin_biaya);
lcd.print(" COIN");
} e
lse
{
lcd.clear();
Serial.print("LOKER SELESAI DIGUNAKAN PADA \r\n");
printDate();
MicroSD();
delay(1000);EEPROM.write(17,0);
EEPROM.write(0,5);
} }
else if(status_locker==5)

```

```

{
if(cnt<6)
{
lcd.setCursor(1,0);
lcd.print("MASUKKAN PASSWORD");
if(key != NO_KEY)
{
lcd.setCursor(x,1);
lcd.print(key);
x++;
j++;
cnt++;
EEPROM.write(j,key);
}
}
else
{
lcd.setCursor(2,3);
lcd.print("CEK PASSWORD (*)");
if(key != NO_KEY)
{
if (key=='*')
{
if(pu1==pU1 && pu2==pU2 && pu3==pU3 &&
pu4==pU4 && pu5==pU5 && pu6==pU6)
{
lcd.clear();
lcd.setCursor(3,1);
lcd.print("PASSWORD BENAR");
delay(3000);lcd.clear();
x=7;
cnt=0;
j=10;
EEPROM.write(0,8);
}
}
else
{
lcd.clear();
EEPROM.write(0,6);

```



```

    }
    }
    }
    }
    }
    else if (status_locker==6)
    {
        lcd.clear();
        lcd.setCursor(3,1);
        lcd.print("PASSWORD SALAH");
        delay (3000);
        cnt_pass++;
        EEPROM.write(17,cnt_pass);
        if(cnt_pass<3)
        {
            lcd.clear();
            cnt=0;
            x=7;
            j=10;
            EEPROM.write(0,5);
        }
        else
        {
            lcd.clear();
            x=7;cnt=0;
            EEPROM.write(17,0);
            j=10;
            EEPROM.write(0,7);
        }
        }
    else if (status_locker==7)
    if(cnt<6)
    {
        lcd.setCursor(1,0);
        lcd.print("MASUKKAN PASSWORD");
        lcd.setCursor(7,1);
        lcd.print("ADMIN");
        if(key != NO_KEY)
        {

```

```

lcd.setCursor(x,2);
lcd.print("*");
x++;
k++;
cnt++;
EEPROM.write(k,key);
}
}
else
{
if(key != NO_KEY)
{
if (key=='#')
{
if(pa1==pA1 && pa2==pA2 && pa3==pA3 &&
pa4==pA4 && pa5==pA5 && pa6==pA6)
{
lcd.clear();
lcd.setCursor(3,1);
lcd.print("PASSWORD BENAR");delay(3000);
lcd.clear();
x=7;
cnt=0;
k=30;
EEPROM.write(0,8);
}
else
{
lcd.clear();
lcd.setCursor(3,1);
lcd.print("HUBUNGI ADMIN");
delay(3000);
lcd.clear();
k=30;
x=7;
cnt=0;
EEPROM.write(0,7);
}
}
}

```

```

}
}
else if (status_locker==8)
{
lcd.setCursor(2,1);
lcd.print("TEKAN (*) UNTUK");
lcd.setCursor(3,2);
lcd.print("MEMBUKA LOKER");
if(key != NO_KEY)
{
if(key=='*')
{
lcd.clear();
lcd.setCursor(1,1);
lcd.print("AMBIL
ANDA");digitalWrite(Solenoid,HIGH);
delay(5000);
lcd.clear();
digitalWrite(Solenoid,LOW);
EEPROM.write(0,9);
delay (1000);
}
}
}
else if (status_locker==9)
{
if(digitalRead(Limit)==LOW)
{
lcd.setCursor(0,1);
lcd.print("TUTUP KEMBALI LOKER");
delay(100);
lcd.clear();
}
}
else
{
lcd.setCursor(4,0);
lcd.print("TERIMA KASIH");
lcd.setCursor(2,1);
lcd.print("TELAH BERKUNJUNG");

```

BARANG

```

lcd.setCursor(4,2);
lcd.print("SEMOGA SUKSES");
lcd.setCursor(3,4);
lcd.print("ANDROMEDA DE-09");
delay(10000);
lcd.clear();
EEPROM.write(0,255);
}
}
}void insert_Coin() {
if(status_locker==1)
{
Coin_sewa=Coin_sewa+1;
EEPROM.write(7,Coin_sewa);
}
else if (status_locker==4)
{
Coin_biaya=Coin_biaya-1;
EEPROM.write(8,Coin_biaya);
} }
void detik(){
if(status_locker == 3)
{
hitung_detik++;
if(hitung_detik ==5)//i menit 60 detik
{
menit++;hitung_detik=0;
EEPROM.write(10,menit);
if(menit == 2)// 1 Coin 30 menit{
{
SetengahJam++;Coin_biaya=Coin_biaya+1;menit=0;
EEPROM.write(10,0);
EEPROM.write(8,Coin_biaya);
if(SetengahJam==2)// i jam 2x SetengahJam
{
Jam++;SetengahJam=0;
EEPROM.write(9,Jam);
}
}
}
}

```

```

}
}}
void decToBcd(){
Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero); //stop Oscillator
Wire.write(decToBcd(second));
Wire.write(decToBcd(minute));
Wire.write(decToBcd(hour));
Wire.write(decToBcd(weekDay));
Wire.write(decToBcd(monthDay));
Wire.write(decToBcd(month));
Wire.write(decToBcd(year));
Wire.write(zero); //start
Wire.endTransmission();
}
byte decToBcd(byte val){
// Konversi Desimal ke Biner
return ( (val/10*16) + (val%10) );
}
byte bcdToDec(byte val) {
// Konversi Biner ke Desimal
return ( (val/16*10) + (val%16) );
}
void printDate(){
Wire.beginTransmission(DS1307_ADDRESS);
Wire.write(zero);
Wire.endTransmission();
Wire.requestFrom(DS1307_ADDRESS, 7);int second =
bcdToDec(Wire.read());
int minute = bcdToDec(Wire.read());
int hour = bcdToDec(Wire.read() & 0b111111);
int weekDay = bcdToDec(Wire.read());
int monthDay = bcdToDec(Wire.read());
int month = bcdToDec(Wire.read());
int year = bcdToDec(Wire.read());
Serial.print(monthDay);
Serial.print("/");
Serial.print(month);
Serial.print("/");

```

```

Serial.print(year);
Serial.print(" ");
Serial.print(hour);
Serial.print(":");
Serial.print(minute);
Serial.print(":");
Serial.println(second);
}
void MicroSD()
{
if(status_locker==2)
{
myFile = SD.open("logger.txt", FILE_WRITE);
//Membuka File test.txt
if (myFile) // jika file tersedia tulis data
{
myFile.print("Mulai Menggunakan");
myFile.print("\n");
myFile.print(monthDay);
myFile.print('/');
myFile.print(month);
myFile.print('/');myFile.print(year);
myFile.print(" (");
myFile.print(hour);
myFile.print(':');
myFile.print(minute);
myFile.print(':');
myFile.print(second);
myFile.print(" )");
myFile.print("\n");
myFile.close();
} }
else if (status_locker==4);
{
myFile = SD.open("logger.txt", FILE_WRITE); //Membuka
File test.txt
if (myFile) // jika file tersedia tulis data
{
myFile.print("Selesai Menggunakan");

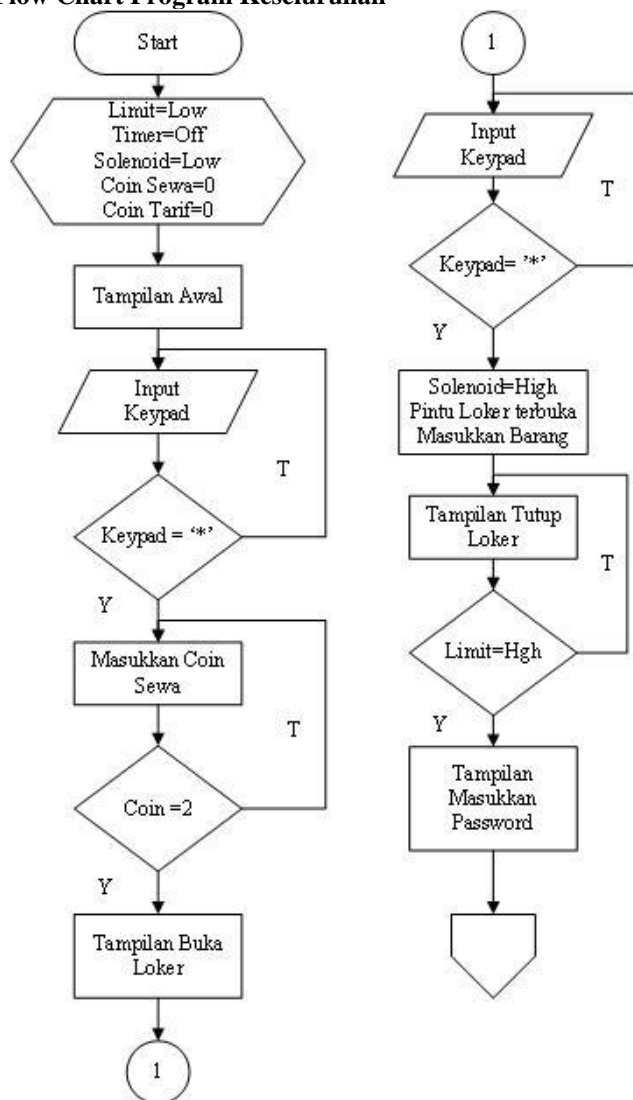
```

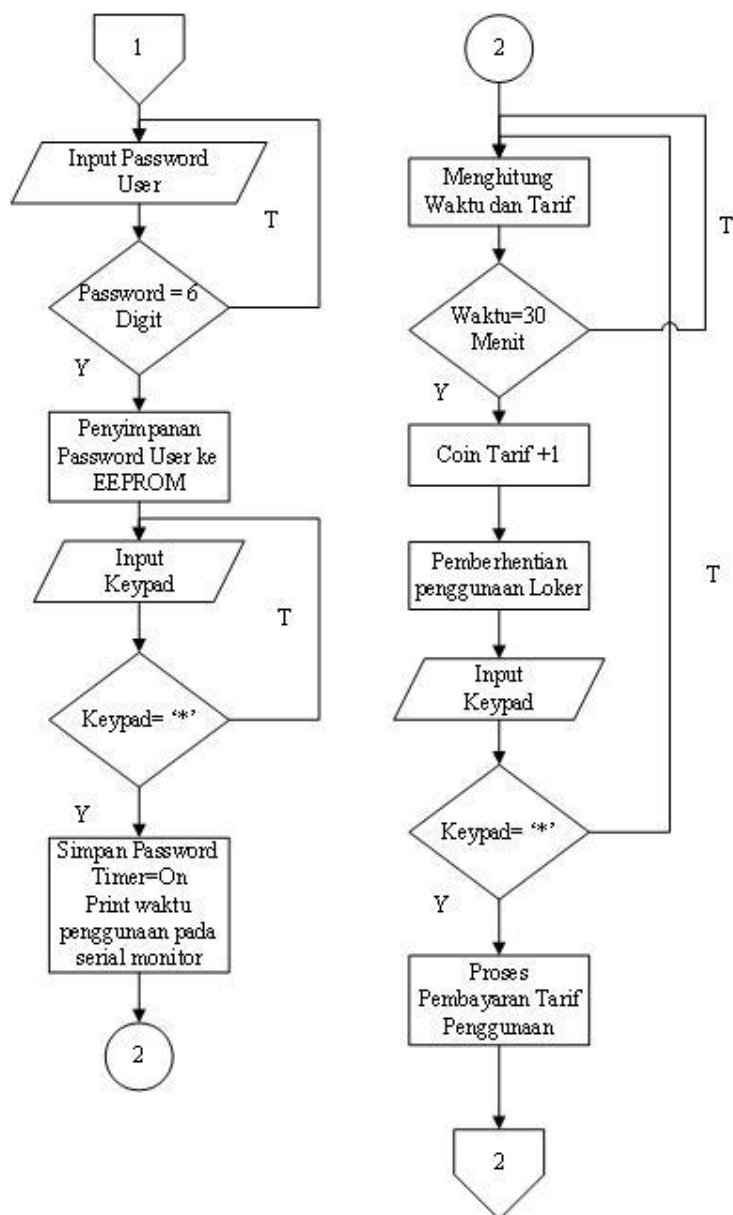
```
myFile.print('\n');
myFile.print(monthDay);
myFile.print('/');
myFile.print(month);
myFile.print('/');
myFile.print(year);
myFile.print(" ");
myFile.print(hour);
myFile.print(':');
myFile.print(minute);
myFile.print(':');
myFile.print(second);
myFile.print(" ");
myFile.print('\n');
myFile.close();
}
}
}
```

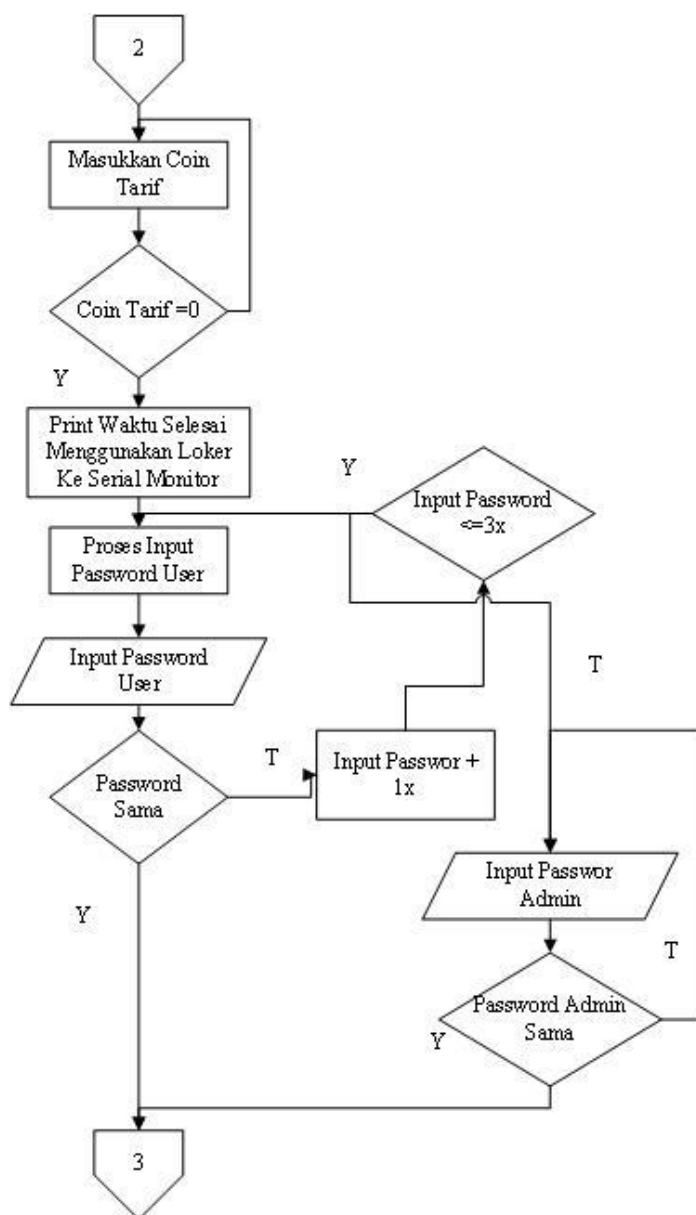
-----Halaman ini sengaja dikosongkan-----

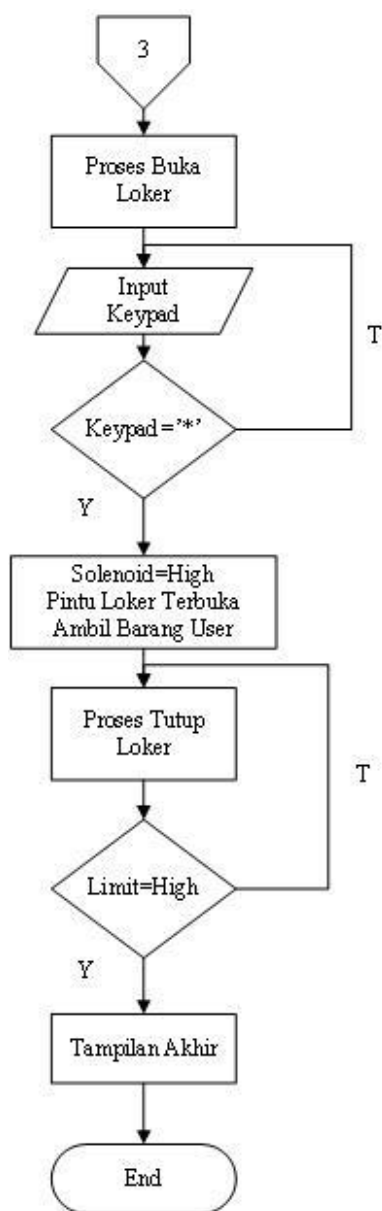


## B. Flow Chart Program Keseluruhan









## C. Datasheet



8-bit AVR Microcontrollers

ATmega328/P

DATASHEET COMPLETE

### Introduction

The Atmel<sup>®</sup> picoPower<sup>®</sup> ATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR<sup>®</sup> enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328/P achieves a throughput close to 1MIPS per MHz. This empowers system designer to optimize the device for power consumption versus processing speed.

### Feature

High Performance, Low Power Atmel<sup>®</sup> AVR<sup>®</sup> 8-Bit Microcontroller Family

- \* Advanced RISC Architecture
  - 131 Powerful Instructions
  - Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20MHz
  - On-chip 2-cycle Multiplier
- \* High Endurance Non-volatile Memory Segments
  - 32KBytes of In-System Self-Programmable Flash program Memory
  - 1KBytes EEPROM
  - 2KBytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data Retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
    - \* In-System Programming by On-chip Boot Program
    - \* True Read-While-Write Operation
  - Programming Lock for Software Security
- \* Atmel<sup>®</sup> QTouch<sup>®</sup> Library Support
  - Capacitive Touch Buttons, Sliders and Wheels
  - QTouch and QMatrix<sup>®</sup> Acquisition
  - Up to 64 sense channels

Atmel-42792B-ATmega328P\_Datasheet\_Complete-11/09/10

- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package
    - Temperature Measurement
  - Two Master/Slave SPI Serial Interface
  - One Programmable Serial USART
  - One Byte-oriented 2-wire Serial Interface (Philips I<sup>2</sup>C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - One On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power save, Power down, Standby, and Extended Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 5.5V
- Temperature Range:
  - -40°C to 105°C
- Speed Grade:
  - 0 - 4MHz @ 1.8 - 5.5V
  - 0 - 10MHz @ 2.7 - 5.5V
  - 0 - 20MHz @ 4.5 - 5.5V
- Power Consumption at 1MHz, 1.8V, 25°C
  - Active Mode: 0.2mA
  - Power-down Mode: 0.1µA
  - Power-save Mode: 0.75µA (including 32kHz RTC)

## 2. Configuration Summary

Features	ATmega328/P
Pin Count	28/32
Flash (Bytes)	32K
SRAM (Bytes)	2K
EEPROM (Bytes)	1K
General Purpose I/O Lines	23
SPI	2
TWI (I <sup>2</sup> C)	1
USART	1
ADC	10-bit 15kSPS
ADC Channels	8
8-bit Timer/Counters	2
16-bit Timer/Counters	1

## 5. Pin Configurations

### 5.1. Pin-out

Figure 5-1. 28-pin PDIP

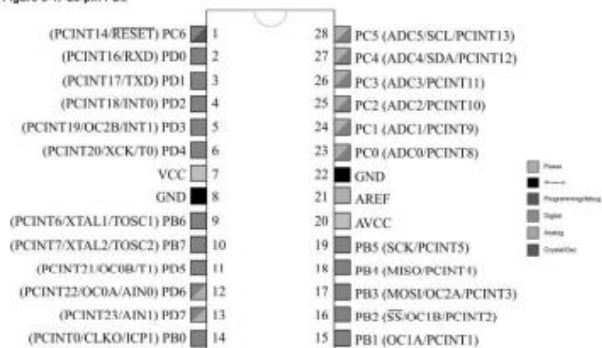
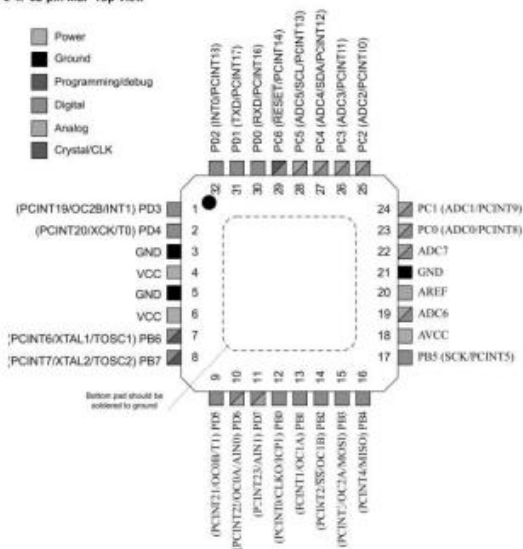




Figure 5-4. 32-pin MLF Top View



## 5.2. Pin Descriptions

### 5.2.1. VCC

Digital supply voltage.

### 5.2.2. GND

Ground.

### 5.2.3. Port B (PB[7:0]) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PD6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.



Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB[7:6] is used as TOSC[2:1] input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

**5.2.4. Port C (PC[5:0])**

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC[5:0] output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**5.2.5. PC6/RESET**

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in the *Alternate Functions of Port C* section.

**5.2.6. Port D (PD[7:0])**

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**5.2.7. AV<sub>CC</sub>**

AV<sub>CC</sub> is the supply voltage pin for the A/D Converter, PC[3:0], and PE[3:2]. It should be externally connected to V<sub>CC</sub>, even if the ADC is not used. If the ADC is used, it should be connected to V<sub>CC</sub> through a low-pass filter. Note that PC[6:4] use digital supply voltage, V<sub>CC</sub>.

**5.2.8. AREF**

AREF is the analog reference pin for the A/D Converter.

**5.2.9. ADC[7:6] (TQFP and VFQFN Package Only)**

In the TQFP and VFQFN package, ADC[7:6] serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

## 6. I/O Multiplexing

Each pin is by default controlled by the PORT as a general purpose I/O and alternatively it can be assigned to one of the peripheral functions.

The following table describes the peripheral signals multiplexed to the PORT I/O pins.

Table 6-1. PORT Function Multiplexing

(32-pin MLF/TQFP) Pin#	(28-pin MLF) Pin#	(28-pin PDIP) Pin#	PA0	EXTINT	PCINT	ADCAIC	OSC	T/C #0	T/C #1	USART 0	DC 0	SPI 0
1	1	5	PD[3]	INT1	PCINT19			OC2B				
2	2	6	PD[4]		PCINT20			T0		XCK0		
4	3	7	VCC									
3	4	8	GND									
6	-	-	VCC									
5	-	-	GND									
7	5	9	PB[6]		PCINT6		XTAL1/ TOSC1					
8	6	10	PB[7]		PCINT7		XTAL2/ TOSC2					
9	7	11	PD[5]		PCINT21			OC0B	T1			
10	8	12	PD[6]		PCINT22	AIN0		OC0A				
11	9	13	PD[7]		PCINT23	AIN1						
12	10	14	PB[0]		PCINT0		CLK0	ICP1				
13	11	15	PB[1]		PCINT1			OC1A				
14	12	16	PB[2]		PCINT2			OC1B				SS0
15	13	17	PB[3]		PCINT3			OC2A				MOSI0
16	14	18	PB[4]		PCINT4							MISO0
17	15	19	PB[5]		PCINT5							CK0
18	16	20	AVCC									
19	-	-	ADC6			ADC6						
20	17	21	AREF									
21	18	22	GND									
22	-	-	ADC7			ADC7						
23	19	13	PC[3]		PCINT8	ADC0						
24	20	24	PC[1]		PCINT9	ADC1						
25	21	25	PC[2]		PCINT10	ADC2						
26	22	26	PC[3]		PCINT11	ADC3						
27	23	27	PC[4]		PCINT12	ADC4					SDA0	
28	24	28	PC[5]		PCINT13	ADC5					SCL0	
29	25	1	PC[0] RESET		PCINT14							

(32-pin MLF/TQFP) Pin#	(28-pin MLF) Pin#	(28-pin PIPD) Pin#	PAD	EXTINT	PCINT	ADC/AC	OSC	T/C #0	T/C #1	USART 0	I2C 0	SPI 0
30	26	2	PD[0]		PCINT16					RXD0		
31	27	3	PD[1]		PCINT17					TXD0		
32	28	4	PD[2]	INT0	PCINT18							

# Manual of CH-926

CH-926 is a multi coin selector, can accept up to 6 kinds of different coins at the same time. This type of coin selector is widely used in Vending machine, Arcade Game, Message chair, and other self-management system. CH-926 is mainly based on material, weight and size to identify coins. We use the most up to date algorithm to design software. Therefore, CH-926 is very stable and accurate even when environment changes such as temperature, and humidity etc... In order to increase the accuracy, we suggest different version of coins use different channel to set up.

## Specifications

Coin diameter : 15mm-32mm	Atmospheric pressure : 86Kpa-106Kpa
Coin thickness : 1.2mm-3.8mm	Working humidity : ≤95%
Working voltage : DC +12V ±10%	Speed : ≤0.6s
Working current : 65mA ±5%	Accuracy rate of identification : 99.5%
Signal output : pulse	

## Features

- a. Capable of accepting all worldwide Coins and Tokens.
- b. Intelligent CPU software control, and high accuracy.
- c. Self-programming without PC.
- d. Accept 1-6 different kinds of coins at the same time.
- e. Free to set up pulses' output.
- f. Prevent not only electric shock but also electromagnetic interference.
- g. Automatic self-test for problems.

# Manual of CH-926

## The Process of Setup for Parameters

1. Press the "Add" and "minus" buttons at the same time for about three seconds, then the letter "A" will appear from the LED display.
2. Press the "setup" button once, and the letter "E" will appear. Next, use the buttons to choose how many kinds of coins you would like to use; then press the "setup" button again to finish.
3. The letter "H" will appear after pressing the button. Use the "Add" and "minus" buttons to choose how many samples you would like to insert later. Next press the "setup" button again to finish.
4. The letter "P" will appear after pressing the button. Use the "Add" and "minus" buttons to choose the amount of output's signals/pulses you want. The quantity limited is 50 times. Next, press the "setup" button to finish.
5. The letter "F" will appear after pressing the button. Use the "Add" and "minus" buttons to choose accuracy. The value is from 1-30, and 1 is the most accurate. Normally, 5-10 will be fine. Next, press the "setup" button to finish.
6. So far, you have successfully set up the first coin, please repeat all above procedures until you have set up all the coins. The letter "A" will appear again after all above procedures are finished.
7. Press the "setup" button, and the letter "E" will appear. Finally, turn off and turn on the power. The setup will be stored.

---

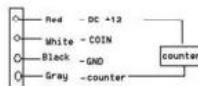
You can start sampling after the setup is finished. Please choose at least 20 coins. The sampling process will affect the accuracy of coin selector.

---

## Sampling

1. Press the "setup" button, then letter "A" will appear from the LED display.
2. Press the "setup" button again, then letter "A1" will appear. Next, start to insert sample coins. The LED display will show how many coins you insert. The letter "A1" will appear again after finished.
3. Press the "setup" button again, then the letter "A2" will appear. Next, Start to sample 2nd coin, and repeat No. 1 and No. 2 until all the coins are set up.
4. After finished the sampling, press the "setup" button. The letter "A" will appear, then turn off and turn on the power. Now you can start to use it.

## Connections' manual

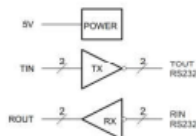


**MAX232x Dual EIA-232 Drivers/Receivers****1 Features**

- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Operates From a Single 5-V Power Supply With 1.0- $\mu$ F Charge-Pump Capacitors
- Operates up to 120 kbit/s
- Two Drivers and Two Receivers
- $\pm 30$ -V Input Levels
- Low Supply Current: 8 mA Typical
- ESD Protection Exceeds JESD 22
  - 2000-V Human-Body Model (A114-A)
- Upgrade With Improved ESD (15-kV HBM) and 0.1- $\mu$ F Charge-Pump Capacitors is Available With the MAX202 Device

**2 Applications**

- TIA/EIA-232-F
- Battery-Powered Systems
- Terminals
- Modems
- Computers

**4 Simplified Schematic****3 Description**

The MAX232 device is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept  $\pm 30$ -V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels.

**Device Information<sup>(1)</sup>**

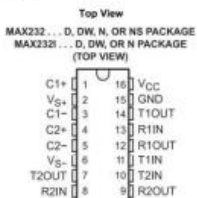
ORDER NUMBER	PACKAGE (PIN)	BODY SIZE
MAX232x	SOIC (16)	9.90 mm × 3.81 mm
	SOIC (16)	10.30 mm × 7.50 mm
	PDIP (16)	19.30 mm × 6.35 mm
	SOIP (16)	10.3 mm × 6.30 mm

(1) For all available packages, see the orderable addendum at the end of the datasheet.



An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

## 6 Pin Configuration and Functions



Pin Functions

NAME	PIN NO.	TYPE	DESCRIPTION
C1+	1	—	Positive lead of C1 capacitor
VS+	2	O	Positive charge pump output for storage capacitor only
C1-	3	—	Negative lead of C1 capacitor
C2+	4	—	Positive lead of C2 capacitor
C2-	5	—	Negative lead of C2 capacitor
VS-	6	O	Negative charge pump output for storage capacitor only
T2OUT, T1OUT	7, 14	O	RS232 line data output (to remote RS232 system)
R2IN, R1IN	8, 13	I	RS232 line data input (from remote RS232 system)
R2OUT, R1OUT	9, 12	O	Logic data output (to UART)
T2IN, T1IN	10, 11	I	Logic data input (from UART)
GND	15	—	Ground
VCC	16	—	Supply Voltage. Connect to external 5V power supply



## 7 Specifications

### 7.1 Absolute Maximum Ratings<sup>(1)</sup>

over operating free-air temperature range (unless otherwise noted)

		MIN	MAX	UNIT
$V_{CC}$	Input supply voltage range <sup>(2)</sup>	-0.3	6	V
$V_{O+}$	Positive output supply voltage range	$V_{CC} - 0.3$	15	V
$V_{O-}$	Negative output supply voltage range	-0.3	-15	V
$V_I$	Input voltage range	T1IN, T2IN -0.3	$V_{CC} + 0.3$	V
		R1IN, R2IN	$\pm 30$	
$V_O$	Output voltage range	T1OUT, T2OUT $V_{O-} - 0.3$	$V_{O+} + 0.3$	V
		R1OUT, R2OUT	-0.3	$V_{O+} + 0.3$
	Short-circuit duration	T1OUT, T2OUT	Unlimited	
$T_J$	Operating virtual junction temperature		150	°C

(1) Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under Recommended Operating Conditions is not implied. Exposure to absolute maximum-rated conditions for extended periods may affect device reliability.

(2) All voltages are with respect to network GND.

### 7.2 Handling Ratings

		MIN	MAX	UNIT
$T_{stg}$	Storage temperature range	-65	150	°C
$V_{ESD}$	Electrostatic discharge	Human body model (HBM), per ANSI/ESDA/JEDEC JS-001, all pins <sup>(1)</sup> Charged device model (CDM), per JEDEC specification JESD22-C101, all pins <sup>(2)</sup>	0 1000	V

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.

(2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

### 7.3 Recommended Operating Conditions

		MIN	NOM	MAX	UNIT
$V_{CC}$	Supply voltage	4.5	5	5.5	V
$V_{IH}$	High-level input voltage (T1IN, T2IN)	2			V
$V_{IL}$	Low-level input voltage (T1IN, T2IN)			0.8	V
R1IN, R2IN	Receiver input voltage			$\pm 30$	V
$T_A$	Operating free-air temperature	MAX232 MAX232I	0 -40	70 85	°C

### 7.4 Thermal Information

THERMAL METRIC <sup>(1)</sup>	MAX232HD	MAX232HDW	MAX232HI	MAX232HIS	UNIT
	SOIC	SOIC wide	PDP	SOP	
$R_{JA}$	18 PINS	18 PINS	16 PINS	16 PINS	°C/W
	73	57	67	64	

(1) For more information about traditional and new thermal metrics, see the IC Package Thermal Metrics application report (SPRA063).

### 7.5 Electrical Characteristics — Device

over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Figure 6)

PARAMETER	TEST CONDITIONS <sup>(1)</sup>	MIN	TYP <sup>(2)</sup>	MAX	UNIT
$I_{CC}$	Supply current	$V_{CC} = 5.5V$ , all outputs open, $T_A = 25^\circ C$	8	10	mA

(1) Test conditions are C1–C4 = 1  $\mu F$  at  $V_{CC} = 5V \pm 0.5V$ .

(2) All typical values are at  $V_{CC} = 5V$ , and  $T_A = 25^\circ C$ .

## 7.6 Electrical Characteristics — Driver

over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted)

PARAMETER	TEST CONDITIONS <sup>(1)</sup>	MIN	TYP <sup>(2)</sup>	MAX	UNIT
$V_{OH}$ High-level output voltage	T1OUT, T2OUT $R_L = 3\text{ k}\Omega$ to GND	5	7		V
$V_{OL}$ Low-level output voltage <sup>(3)</sup>	T1OUT, T2OUT $R_L = 3\text{ k}\Omega$ to GND		-7	-5	V
$t_O$ Output resistance	T1OUT, T2OUT $V_{SA} = V_{SD} = 0, V_O = \pm 2\text{ V}$	300			$\Omega$
$I_{OZ}^{(4)}$ Short-circuit output current	T1OUT, T2OUT $V_{CC} = 5.5\text{ V}, V_O = 0\text{ V}$		$\pm 10$		mA
$I_{IN}$ Short-circuit input current	T1IN, T2IN $V_I = 0$			200	$\mu\text{A}$

(1) Test conditions are  $C_1$ - $C_4 = 1\text{ }\mu\text{F}$  at  $V_{CC} = 5\text{ V} \pm 0.5\text{ V}$ .

(2) All typical values are at  $V_{CC} = 5\text{ V}, T_A = 25^\circ\text{C}$ .

(3) The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

(4) Not more than one output should be shorted at a time.

## 7.7 Electrical Characteristics — Receiver

over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted)

PARAMETER	TEST CONDITIONS <sup>(1)</sup>	MIN	TYP <sup>(2)</sup>	MAX	UNIT
$V_{OH}$ High-level output voltage	R1OUT, R2OUT $I_{OH} = -1\text{ mA}$	3.5			V
$V_{OL}$ Low-level output voltage <sup>(3)</sup>	R1OUT, R2OUT $I_{OL} = 3.2\text{ mA}$			0.4	V
$V_{IT+}$ Receiver positive-going input threshold voltage	R1IN, R2IN $V_{CC} = 5\text{ V}, T_A = 25^\circ\text{C}$		1.7	2.4	V
$V_{IT-}$ Receiver negative-going input threshold voltage	R1IN, R2IN $V_{CC} = 5\text{ V}, T_A = 25^\circ\text{C}$	0.8	1.2		V
$V_{HY}$ Input hysteresis voltage	R1IN, R2IN $V_{CC} = 5\text{ V}$	0.2	0.5	1	V
$r_i$ Receiver input resistance	R1IN, R2IN $V_{CC} = 5\text{ V}, T_A = 25^\circ\text{C}$	3	5	7	k $\Omega$

(1) Test conditions are  $C_1$ - $C_4 = 1\text{ }\mu\text{F}$  at  $V_{CC} = 5\text{ V} \pm 0.5\text{ V}$ .

(2) All typical values are at  $V_{CC} = 5\text{ V}, T_A = 25^\circ\text{C}$ .

(3) The algebraic convention, in which the least-positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

## 7.8 Switching Characteristics

over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted)

PARAMETER	TEST CONDITIONS <sup>(1)</sup>	MIN	TYP <sup>(2)</sup>	MAX	UNIT
SR Driver slew rate	$R_L = 3\text{ k}\Omega$ to $7\text{ k}\Omega$ , see Figure 4			30	V/ $\mu\text{s}$
SR(t) Driver transition region slew rate	see Figure 5		3		V/ $\mu\text{s}$
Data rate	One TOUT switching		120		kb/s
$t_{PLH}$ Receiver propagation delay time, low- to high-level output	TTL load, see Figure 3		500		ns
$t_{PLH}$ Receiver propagation delay time, high- to low-level output	TTL load, see Figure 3		500		ns

(1) Test conditions are  $C_1$ - $C_4 = 1\text{ }\mu\text{F}$  at  $V_{CC} = 5\text{ V} \pm 0.5\text{ V}$ .

## GENERAL DESCRIPTION

The DS1307 serial real-time clock (RTC) is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially through an I<sup>2</sup>C, bidirectional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeeping operation continues while the part operates from the backup supply.

## FEATURES

- Real-Time Clock (RTC) Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the week, and Year with Leap-Year Compensation Valid Up to 2100
- 56-Byte, Battery-Backed, Nonvolatile (NV) RAM for Data Storage
- I<sup>2</sup>C Serial Interface
- Programmable Square-Wave Output Signal
- Automatic Power-Fail Detect and Switch Circuitry
- Consumes Less than 500nA in Battery-Backup Mode with Oscillator Running
- Optional Industrial Temperature Range: -40°C to +85°C
- Available in 8-Pin Plastic DIP or SO
- Underwriters Laboratory (UL) Recognized

Typical Operating Circuit and Pin Configurations appear at end of data sheet.

## ORDERING INFORMATION

PART	TEMP RANGE	VOLTAGE (V)	PIN-PACKAGE	TOP MARK*
DS1307	0°C to +70°C	5.0	8 PDIP (300 mils)	DS1307
DS1307+	0°C to +70°C	5.0	8 PDIP (300 mils)	DS1307
DS1307N	-40°C to +85°C	5.0	8 PDIP (300 mils)	DS1307N
DS1307N+	-40°C to +85°C	5.0	8 PDIP (300 mils)	DS1307N
DS1307Z	0°C to +70°C	5.0	8 SO (150 mils)	DS1307
DS1307Z+	0°C to +70°C	5.0	8 SO (150 mils)	DS1307
DS1307ZN	-40°C to +85°C	5.0	8 SO (150 mils)	DS1307N
DS1307ZN+	-40°C to +85°C	5.0	8 SO (150 mils)	DS1307N
DS1307Z/T&R	0°C to +70°C	5.0	8 SO (150 mils) Tape and Reel	DS1307
DS1307Z+/T&R	0°C to +70°C	5.0	8 SO (150 mils) Tape and Reel	DS1307
DS1307ZN/T&R	-40°C to +85°C	5.0	8 SO (150 mils) Tape and Reel	DS1307N
DS1307ZN+/T&R	-40°C to +85°C	5.0	8 SO (150 mils) Tape and Reel	DS1307N

+ Denotes a lead-free/RoHS-compliant device.

\* A "+" anywhere on the top mark indicates a lead-free device.

Note: Some revisions of this device may incorporate deviations from published specifications known as errata. Multiple revisions of any device may be simultaneously available through various sales channels. For information about device errata, click here: [www.maxim-ic.com/errata](http://www.maxim-ic.com/errata)

**ABSOLUTE MAXIMUM RATINGS**

Voltage Range on Any Pin Relative to Ground.....	-0.5V to +7.0V
Operating Temperature Range (Noncondensing)	
Commercial.....	0°C to +70°C
Industrial.....	-40°C to +85°C
Storage Temperature Range.....	-55°C to +125°C
Soldering Temperature (DIP, leads).....	+260°C for 10 seconds
Soldering Temperature (surface mount).....	See JPC/JEDEC Standard J-STD-020

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to the absolute maximum rating conditions for extended periods may affect device reliability.

**RECOMMENDED DC OPERATING CONDITIONS**

( $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ ,  $T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ .) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	$V_{CC}$		4.5	5.0	5.5	V
Logic 1 Input	$V_{IH}$		2.2		$V_{CC} + 0.3$	V
Logic 0 Input	$V_{IL}$		-0.3		+0.8	V
$V_{BAT}$ Battery Voltage	$V_{BAT}$		2.0	3	3.5	V

**DC ELECTRICAL CHARACTERISTICS**

( $V_{CC} = 4.5\text{V}$  to  $5.5\text{V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ ,  $T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ .) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Input Leakage (SCL)	$I_{IJ}$		-1		1	$\mu\text{A}$
I/O Leakage (SDA, SQW/OUT)	$I_{IO}$		-1		1	$\mu\text{A}$
Logic 0 Output ( $I_{OL} = 5\text{mA}$ )	$V_{OL}$				0.4	V
Active Supply Current ( $f_{SCL} = 100\text{kHz}$ )	$I_{CCA}$				1.5	mA
Standby Current	$I_{ECS}$	(Note 3)			200	$\mu\text{A}$
$V_{BAT}$ Leakage Current	$I_{BATLEK}$			5	50	nA
Power-Fail Voltage ( $V_{BAT} = 3.0\text{V}$ )	$V_{PF}$		1.216 N $V_{BAT}$	1.25 N $V_{BAT}$	1.284 N $V_{BAT}$	V

**DC ELECTRICAL CHARACTERISTICS**

( $V_{CC} = 0\text{V}$ ,  $V_{BAT} = 3.0\text{V}$ ,  $T_A = 0^\circ\text{C}$  to  $+70^\circ\text{C}$ ,  $T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ .) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
$V_{BAT}$ Current (OSC ON); SQW/OUT OFF	$I_{BAT1}$			300	500	nA
$V_{BAT}$ Current (OSC ON); SQW/OUT ON (32kHz)	$I_{BAT2}$			480	800	nA
$V_{BAT}$ Data-Retention Current (Oscillator Off)	$I_{BATDR}$			10	100	nA

**WARNING:** Negative undershoots below -0.3V while the part is in battery-backed mode may cause loss of data.

**AC ELECTRICAL CHARACTERISTICS**(V<sub>CC</sub> = 4.5V to 5.5V; T<sub>A</sub> = 0°C to +70°C, T<sub>A</sub> = -40°C to +85°C.)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f <sub>SCL</sub>		0	100		kHz
Bus Free Time Between a STOP and START Condition	t <sub>BUF</sub>		4.7			μs
Hold Time (Repeated) START Condition	t <sub>HOLD,STA</sub>	(Note 4)	4.0			μs
LOW Period of SCL Clock	t <sub>LOW</sub>		4.7			μs
HIGH Period of SCL Clock	t <sub>HIGH</sub>		4.0			μs
Setup Time for a Repeated START Condition	t <sub>SET,STA</sub>		4.7			μs
Data Hold Time	t <sub>HOLD,DAT</sub>		0			μs
Data Setup Time	t <sub>SET,DAT</sub>	(Notes 5, 6)	250			ns
Rise Time of Both SDA and SCL Signals	t <sub>R</sub>			1000		ns
Fall Time of Both SDA and SCL Signals	t <sub>F</sub>			300		ns
Setup Time for STOP Condition	t <sub>SET,STO</sub>		4.7			μs

**CAPACITANCE**(T<sub>A</sub> = +25°C)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Pin Capacitance (SDA, SCL)	C <sub>IO</sub>			10		pF
Capacitance Load for Each Bus Line	C <sub>B</sub>	(Note 7)		400		pF

**Note 1:** All voltages are referenced to ground.**Note 2:** Limits at -40°C are guaranteed by design and are not production tested.**Note 3:** I<sub>CCS</sub> specified with V<sub>CC</sub> = 5.0V and SDA, SCL = 5.0V.**Note 4:** After this period, the first clock pulse is generated.**Note 5:** A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the V<sub>BI(MIN)</sub> of the SCL signal) to bridge the undefined region of the falling edge of SCL.**Note 6:** The maximum t<sub>HOLD,DAT</sub> only has to be met if the device does not stretch the LOW period (t<sub>LOW</sub>) of the SCL signal.**Note 7:** C<sub>B</sub>—total capacitance of one bus line in pF.

**PIN DESCRIPTION**

PIN	NAME	FUNCTION
1	X1	Connections for Standard 32.768kHz Quartz Crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance ( $C_L$ ) of 12.5pF. X1 is the input to the oscillator and can optionally be connected to an external 32.768kHz oscillator. The output of the internal oscillator, X2, is floated if an external oscillator is connected to X1.  <b>Note:</b> For more information on crystal selection and crystal layout considerations, refer to <i>Application Note 28: Crystal Considerations with Dallas Real-Time Clocks</i> .
2	X2	
3	V <sub>BAT</sub>	Backup Supply Input for Any Standard 3V Lithium Cell or Other Energy Source. Battery voltage must be held between the minimum and maximum limits for proper operation. Diodes in series between the battery and the V <sub>BAT</sub> pin may prevent proper operation. If a backup supply is not required, V <sub>BAT</sub> must be grounded. The nominal power-fail trip point ( $V_{PF}$ ) voltage at which access to the RTC and user RAM is denied is set by the internal circuitry as $1.25 \times V_{BAT}$ nominal. A lithium battery with 48mAh or greater will back up the DS1307 for more than 10 years in the absence of power at +25°C.  UL recognized to ensure against reverse charging current when used with a lithium battery. Go to: <a href="http://www.maxim-ic.com/qa/info/ul/">www.maxim-ic.com/qa/info/ul/</a> .
4	GND	Ground
5	SDA	Serial Data Input/Output. SDA is the data input/output for the I <sup>2</sup> C serial interface. The SDA pin is open drain and requires an external pullup resistor.
6	SCL	Serial Clock Input. SCL is the clock input for the I <sup>2</sup> C interface and is used to synchronize data movement on the serial interface.
7	SWQ/OUT	Square Wave/Output Driver. When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square-wave frequencies (1Hz, 4kHz, 8kHz, 32kHz). The SQW/OUT pin is open drain and requires an external pullup resistor. SQW/OUT operates with either V <sub>CC</sub> or V <sub>BAT</sub> applied.
8	V <sub>CC</sub>	Primary Power Supply. When voltage is applied within normal limits, the device is fully accessible and data can be written and read. When a backup supply is connected to the device and V <sub>CC</sub> is below V <sub>TP</sub> , read and writes are inhibited. However, the timekeeping function continues unaffected by the lower input voltage.

**DETAILED DESCRIPTION**

The DS1307 is a low-power clock/calendar with 56 bytes of battery-backed SRAM. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The DS1307 operates as a slave device on the I<sup>2</sup>C bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V<sub>CC</sub> falls below  $1.25 \times V_{BAT}$ , the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out-of-tolerance system. When V<sub>CC</sub> falls below V<sub>BAT</sub>, the device switches into a low-current battery-backup mode. Upon power-up, the device switches from battery to V<sub>CC</sub> when V<sub>CC</sub> is greater than V<sub>BAT</sub> + 0.2V and recognizes inputs when V<sub>CC</sub> is greater than  $1.25 \times V_{BAT}$ . The block diagram in Figure 1 shows the main elements of the serial RTC.

### OSCILLATOR CIRCUIT

The DS1307 uses an external 32.768kHz crystal. The oscillator circuit does not require any external resistors or capacitors to operate. Table 1 specifies several crystal parameters for the external crystal. Figure 1. shows a functional schematic of the oscillator circuit. If using a crystal with the specified characteristics, the startup time is usually less than one second.

### CLOCK ACCURACY

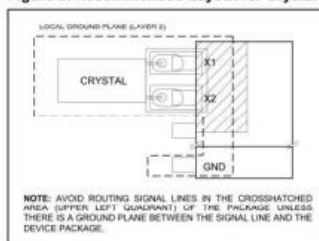
The accuracy of the clock is dependent upon the accuracy of the crystal and the accuracy of the match between the capacitive load of the oscillator circuit and the capacitive load for which the crystal was trimmed. Additional error will be added by crystal frequency drift caused by temperature shifts. External circuit noise coupled into the oscillator circuit may result in the clock running fast. Refer to *Application Note 58: Crystal Considerations with Dallas Real-Time Clocks* for detailed information.

**Table 1. Crystal Specifications\***

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS
Nominal Frequency	$f_0$		32.768		kHz
Series Resistance	ESR			45	k $\Omega$
Load Capacitance	$C_L$		12.5		pF

\*The crystal, traces, and crystal input pins should be isolated from RF generating signals. Refer to Application Note 58: Crystal Considerations for Dallas Real-Time Clocks for additional specifications.

**Figure 2. Recommended Layout for Crystal**



### RTC AND RAM ADDRESS MAP

Table 2 shows the address map for the DS1307 RTC and RAM registers. The RTC registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multibyte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

**CLOCK AND CALENDAR**

The time and calendar information is obtained by reading the appropriate register bytes. Table 2 shows the RTC registers. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the BCD format. The day-of-week register increments at midnight. Values that correspond to the day of week are user-defined but must be sequential (i.e., if 1 equals Sunday, then 2 equals Monday, and so on.) Illogical time and date entries result in undefined operation. Bit 7 of Register 0 is the clock halt (CH) bit. When this bit is set to 1, the oscillator is disabled. When cleared to 0, the oscillator is enabled.

**Note that the initial power-on state of all registers is not defined. Therefore, it is important to enable the oscillator (CH bit = 0) during initial configuration.**

The DS1307 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12-hour or 24-hour mode-select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24-hour mode, bit 5 is the second 10-hour bit (20 to 23 hours). The hours value must be re-entered whenever the 12/24-hour mode bit is changed.

When reading or writing the time and date registers, secondary (user) buffers are used to prevent errors when the internal registers update. When reading the time and date registers, the user buffers are synchronized to the internal registers on any I<sup>2</sup>C START. The time information is read from these secondary registers while the clock continues to run. This eliminates the need to re-read the registers in case the internal registers update during a read. The divider chain is reset whenever the seconds register is written. Write transfers occur on the I<sup>2</sup>C acknowledge from the DS1307. Once the divider chain is reset, to avoid rollover issues, the remaining time and date registers must be written within one second.

**Table 2. Timekeeper Registers**

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00H	CH	10 Seconds			Seconds			Seconds		00–59
01H	0	10 Minutes			Minutes			Minutes		00–59
02H	0	12	10 Hour	10 Hour	Hours			Hours	1–12 +AM/PM 00–23	
		24	PM/ AM							
03H	0	0	0	0	0	DAY			Day	01–07
04H	0	0	10 Date		Date		Date		Date	01–31
05H	0	0	0	10 Month	Month			Month	01–12	
06H	10 Year				Year			Year	00–99	
07H	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08H-3FH									RAM 56 x 8	00H-FFH

0 = Always reads back as 0.



**CONTROL REGISTER**

The DS1307 control register is used to control the operation of the SQW/OUT pin.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

**Bit 7: Output Control (OUT).** This bit controls the output level of the SQW/OUT pin when the square-wave output is disabled. If SQWE = 0, the logic level on the SQW/OUT pin is 1 if OUT = 1 and is 0 if OUT = 0.

**Bit 4: Square-Wave Enable (SQWE).** This bit, when set to logic 1, enables the oscillator output. The frequency of the square-wave output depends upon the value of the RS0 and RS1 bits. With the square-wave output set to 1Hz, the clock registers update on the falling edge of the square wave.

**Bits 1, 0: Rate Select (RS1, RS0).** These bits control the frequency of the square-wave output when the square-wave output has been enabled. The following table lists the square-wave frequencies that can be selected with the RS bits.

RS1	RS0	SQW/OUT OUTPUT	SQWE	OUT
0	0	1Hz	1	X
0	1	4.096kHz	1	X
1	0	8.192kHz	1	X
1	1	32.768kHz	1	X
X	X	0	0	0
X	X	1	0	1

## I<sup>2</sup>C DATA BUS

The DS1307 supports the I<sup>2</sup>C protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are referred to as slaves. The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1307 operates as a slave on the I<sup>2</sup>C bus.

Figures 3, 4, and 5 detail how data is transferred on the I<sup>2</sup>C bus.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is high will be interpreted as control signals.

Accordingly, the following bus conditions have been defined:

**Bus not busy:** Both data and clock lines remain HIGH.

**Start data transfer:** A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

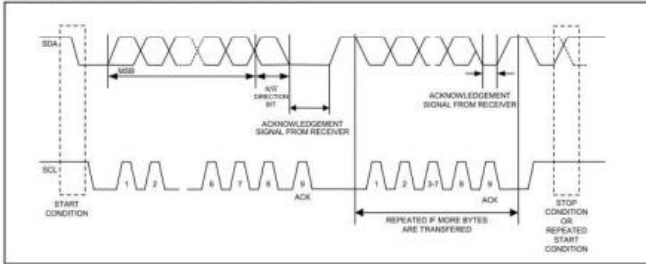
**Stop data transfer:** A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

**Data valid:** The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the I<sup>2</sup>C bus specifications a standard mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1307 operates in the standard mode (100kHz) only.

**Acknowledge:** Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

Figure 3. Data Transfer on I<sup>2</sup>C Serial Bus

Depending upon the state of the  $R/\overline{W}$  bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.
2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

The master device generates all the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

The DS1307 may operate in the following two modes:

1. **Slave Receiver Mode (Write Mode):** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Hardware performs address recognition after reception of the slave address and direction bit (see Figure 4). The slave address byte is the first byte received after the master generates the START condition. The slave address byte contains the 7-bit DS1307 address, which is 1101000, followed by the direction bit (R/W), which for a write is 0. After receiving and decoding the slave address byte, the DS1307 outputs an acknowledge on SDA. After the DS1307 acknowledges the slave address + write bit, the master transmits a word address to the DS1307. This sets the register pointer on the DS1307, with the DS1307 acknowledging the transfer. The master can then transmit zero or more bytes of data with the DS1307 acknowledging each byte received. The register pointer automatically increments after each data byte are written. The master will generate a STOP condition to terminate the data write.
2. **Slave Transmitter Mode (Read Mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the direction bit will indicate that the transfer direction is reversed. The DS1307 transmits serial data on SDA while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (see Figure 5). The slave address byte is the first byte received after the START condition is generated by the master. The slave address byte contains the 7-bit DS1307 address, which is 1101000, followed by the direction bit (R/W), which is 1 for a read. After receiving and decoding the slave address the DS1307 outputs an acknowledge on SDA. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The register pointer automatically increments after each byte are read. The DS1307 must receive a Not Acknowledge to end a read.

Figure 4. Data Write—Slave Receiver Mode

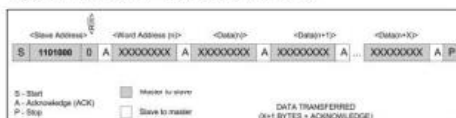
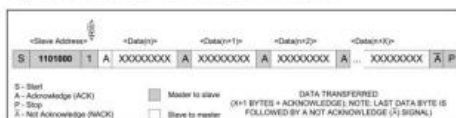
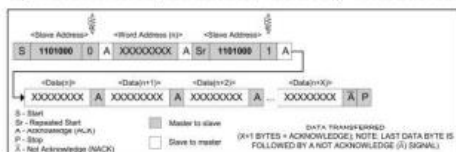
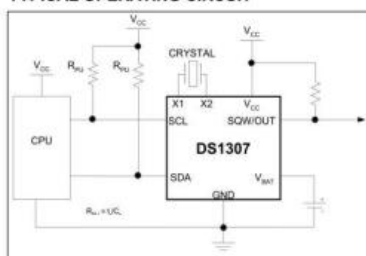
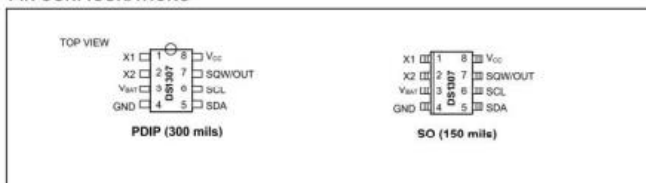


Figure 5. Data Read—Slave Transmitter Mode



**Figure 6. Data Read (Write Pointer, Then Read)—Slave Receive and Transmit****TYPICAL OPERATING CIRCUIT****PIN CONFIGURATIONS**

-----Halaman ini sengaja dikosongkan-----

## **RIWAYAT HIDUP**



Nama : Akbar Riski Darrani  
TTL : Sumenep, 11-03- 1996  
Jenis Kelamin : Laki-Laki  
Agama : Islam  
Alamat : Jln.Raya Guluk-Guluk  
RT 002 RW 001 Kec.  
Guluk-Guluk Kab  
Sumenep  
Telp/HP : 085854979777  
E-Mail : akbar 11maret  
@gmail.com

### **RIWAYAT PENDIDIKAN**

1. 2002 – 2008 : SD Negeri Bataal Barat 1
2. 2008 – 2011 : SMP Negeri 1 Guluk-Guluk
3. 2011 – 2014 : SMA Negeri 1 Sumenep
4. 2014 – 2017 : Departemen Teknik Elektro Otomasi Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

### **PENGALAMAN KERJA**

1. Kerja Praktek di Balai Riset dan Standardisasi Industri Surabaya

### **PENGALAMAN ORGANISASI**

1. Panitia Industrial Automation and Robotic Competition-IARC 2016

-----Halaman ini sengaja dikosongkan-----